Satellite Image Deconvolution Using Complex Wavelet Packets

André Jalobeanu, Laure Blanc-Féraud, Josiane Zerubia

INRIA

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Satellite image deconvolution using complex wavelet packets

A. Jalobeanu — L. Blanc-Féraud — J. Zerubia

N° 3955

June 2000

THÈME 3

ISSN 0249-6399 ISRN INRIA/RR--3955--FR+ENG

INRIA
SOPHIA ANTIPOLIS

Satellite image deconvolution using complex wavelet packets

A. Jalobeanu , L. Blanc-Féraud , J. Zerubia

Thème 3 —Interaction homme-machine,
images, données, connaissances
Projet Ariana

Rapport de recherche n° 3955 —June 2000 —116 pages

Abstract: The deconvolution of blurred and noisy satellite images is an ill-posed inverse problem. The direct inversion leads to unacceptable noise amplification. Usually, either the problem is regularized during the inversion process, or the noise is filtered after deconvolution and decomposition in the wavelet transform domain. Herein, we have developed the second solution, by thresholding the coefficients of a new complex wavelet packet transform ; the thresholding functions are automatically estimated. The use of complex wavelet packets enables translation invariance, and takes into account the directions, while remaining of complexity O(N).

The obtained results exhibit both correctly restored textures and a high SNR in homogeneous areas. Compared to concurrent algorithms, the proposed method is faster, rotation invariant and takes into account the directions of the details and textures of the image to restore them better.

The images deconvolved this way can be used as they are (the restoration step proposed here can be directly inserted in the acquisition chain). But they also can provide a starting point of an adaptive regularization method, enabling one to obtain sharper edges.

Key-words: Deconvolution, Estimation techniques, Complex wavelet packets, Satellite images

Acknowledgements: The authors would like to thank Jérôme Kalifa (from CMAPX, at École Polytechnique) for interesting discussions and Nick Kingsbury (from the Signal Processing Group, Dept. of Eng., University of Cambridge) for complex wavelets source code and collaboration, Peter de Rivaz (same institution) for his kind remarks, and the French Space Agency (CNES) for providing the image of Nîmes (SPOT 5 simulation).

Unité de recherche INRIA Sophia Antipolis
2004, route des Lucioles, B.P. 93, 06902 Sophia Antipolis Cedex (France)
Téléphone : 04 92 38 77 77 - International : +33 4 92 38 77 77 —Fax : 04 92 38 77 65 - International : +33 4 92 38 77 65

Déconvolution d'images satellitaires par paquets d'ondelettes complexes

Résumé : La déconvolution des images satellitaires floues et bruitées est un problème inverse mal posé. L'inversion directe entraîne une amplification inacceptable du bruit. Généralement, soit le problème est régularisé lors de l'inversion, soit le bruit est filtré après déconvolution et décomposition dans le domaine de la transformée en ondelettes. Nous avons developpé dans ce rapport la deuxième solution, en seuillant les coefficients d'une nouvelle transformée en paquets d'ondelettes complexes, les fonctions de seuillage étant estimées de manière automatique. L'utilisation de paquets d'ondelettes complexes rend cette méthode invariante par translation, et tient compte des directions, tout en restant d'une complexité O(N).

Les résultats obtenus présentent à la fois des textures nettes et un très bon rapport signal/bruit dans les zones homogènes. Par rapport aux algorithmes concurrents, la méthode que nous proposons est plus rapide, invariante par rotation, et tient compte de la directionnalité des détails et des textures de l'image pour mieux les restaurer.

Les images déconvoluées de cette manière peuvent être utilisées telles quelles (la restauration peut être intégrée directement dans la chaîne d'acquisition). Mais elles peuvent également constituer le point de départ d'une méthode de régularisation adaptative, permettant d'obtenir des contours plus francs.

Mots-clés : Déconvolution, Techniques d'estimation, Paquets d'ondelettes complexes, Images satellitaires

Remerciements : Les auteurs souhaitent remercier Jérôme Kalifa (CMAPX, École Polytechnique) pour des discussions fructueuses, Nick Kingsbury (Signal Processing Group, Dept. of Eng., University of Cambridge) pour le code source des ondelettes complexes et pour sa collaboration, Peter de Rivaz (même institut) pour ses remarques pertinentes, ainsi que le CNES pour l'image de Nîmes (simulation SPOT 5).

Contents

Notation

Images have $N_x \times N_y$ pixels.

X is a vector, made from the pixels of an image in a lexicographic order.
$X_{i,j}$ is the value of the pixel at column i and line j.
X^n is the n^{th} vector of the chain (X^n).

Let us write $P(X)$ instead of $P(X = x)$, where x is a random vector.

$\mathcal{X}$ is the original image, Y is the observed (corrupted) image.

h is the Point Spread Function (PSF) and has the same dimension as X.

$\mathcal{F}$ stands for the Fourier transform. DCT stands for the Discrete Cosine Transform.

$\mathbf{L}^2(\mathbb{R})$ denotes the class of unidimensional measurable functions f for which f^2 is integrable, and $\mathbf{L}^2(\mathbb{R}^2)$ is the class associated to bidimensional functions.

$\mathrm{l}^2(\mathbb{R})$ denotes the class of unidimensional measurable discrete functions f for which f^2 is integrable, and $\mathrm{l}^2(\mathbb{R}^2)$ is the class associated to bidimensional discrete functions.

Chapter 1

Satellite image deconvolution

1.1 Introduction to satellite image deconvolution

The problem presented here is the reconstruction of a satellite image from blurred and noisy data.

The degradation model is represented by the equation

$$Y = h \star \mathcal{X} + N \tag{1.1}$$

where Y is the observed data, and $\mathcal{X}$ the original image. N is an additive noise and is supposed to be Gaussian, white and stationary. The $\star$ represents a circular convolution. The Point Spread Function (PSF) h is positive, and follows the Shannon property.

We deal with a real satellite image deblurring problem, proposed by the French Space Agency (CNES). This problem is part of a simulation of the future SPOT 5 satellite. Both original and degraded images are provided by CNES. The noise standard deviation σ and the PSF h are also provided ($\sigma = 1.35$ for the example presented in chapter 4, see Fig. 1.1 for h). In this case, h is symmetric and separable with respect to lines and columns, but the formalism presented in this report can be extended to the more general case where these properties are not satisfied.

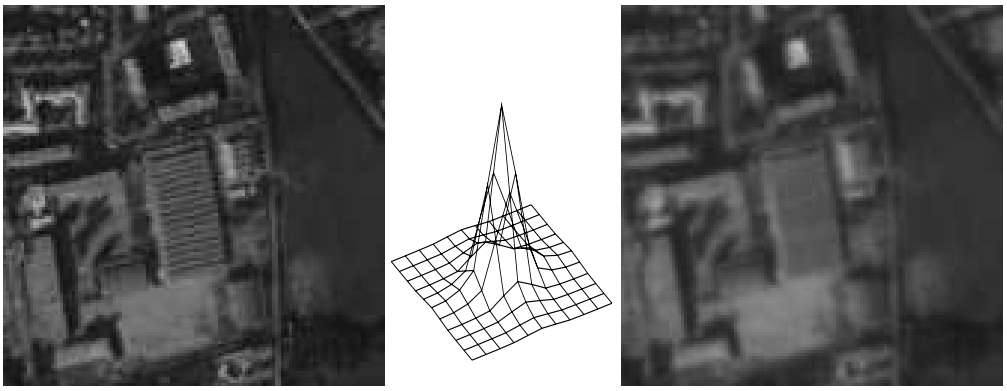


Figure 1.1: *left : original image $\mathcal{X}$ extracted from Nîmes ©CNES, center : PSF h (only a 11×11 area of nonnull coefficients is shown), right : observed image Y*

1.2 Efficient image representations for denoising

The deconvolution problem is an ill-posed inverse problem, because of the noise which contaminates the data. The inversion process strongly amplifies the noise, if no regularization is done. Thus, we have to deconvolve the observed image which means recovering the details, without amplifying the noise.

We expect that the process used to estimate $\mathcal{X}$ from the degraded data Y must preserve the textures, to enable the results to be visually correct. Moreover, the noise must remain small in homogeneous parts. Many methods have been proposed to regularize this problem by introducing a priori constraints on the solution [3, 7, 10, 15, 20, 21, 44]. However, most of them do not preserve the textures, since they are not taken into account in the regularizing model.

To achieve a better deconvolution, a few authors such as Donoho et al. [16, 17, 18], Mallat and Kalifa [29, 30, 31, 32, 38], have proposed to *denoise* the image after a *deconvolution without regularization*. The images are represented using a wavelet or wavelet packet basis, and the denoising process is done in this basis.

A simple inversion of the observation equation (1.1) in the frequency space, consisting of dividing $\mathcal{F}[Y]$ by $\mathcal{F}[h]$, gives an unacceptable noisy solution. To denoise such a solution, an efficient image representation has to be chosen, in order to separate the signal from the noise as much as possible.

1.2.1 Compact representations

A compact representation consists of approximating a signal with few parameters, which can be the coefficients of the decomposition in a given basis. This basis has to be adapted to the structure of the data to be restored. To denoise a signal contaminated by some white noise, the signal has to be decomposed in a basis which transforms it into a few high amplitude coefficients, with the rest of low amplitude. Then, most of the noise is suppressed by thresholding the representation, i.e. by cancelling the smallest coefficients [16]. Donoho and Johnstone [18] have shown that such a thresholding estimator is asymptotically optimal, if the signal representation is *sufficiently compact*.

1.2.2 Noise covariance diagonalization

The noise amplified by the deconvolution process is colored, this will be shown in section 2.1.2. Furthermore, the coefficients of this noise are not independent in the wavelet basis. Therefore one needs to adapt the basis to the covariance properties of the noise. The covariance should be *nearly diagonal* w.r.t. the basis, to decorrelate the noise coefficients as much as possible. The Fourier basis achieves such a diagonalization, but the energy of the signal is not concentrated over a small number of coefficients (the basis vectors are not spatially localized), so the Fourier transform is not suitable for any thresholding method.

A good compromise is made if a *wavelet packet* basis [13] is used, since it nearly realizes the two essential conditions, i.e. the signal representation is sparse, the noise covariance operator is nearly diagonalized (see Fig. 1.2 for an illustration : the signal and the noise are efficiently separated).

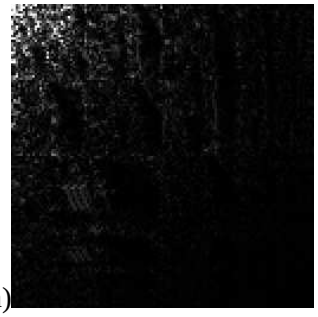
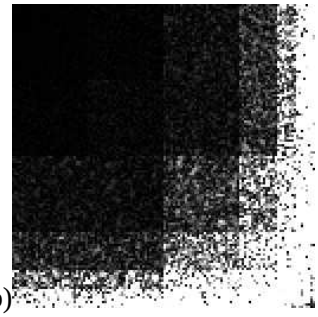
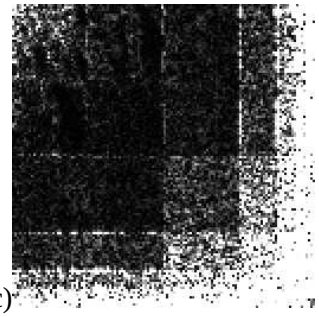
a)  b)  c) 

Figure 1.2: *Representations in a real wavelet packet basis : a) original image $\mathcal{X}$, b) deconvolved noise, c) deconvolved image by nonregularized inversion*

1.2.3 Real wavelet packet thresholding

In chapter 2, we will first detail the method proposed by Kalifa and Mallat, consisting of using wavelet packets within a minimax estimation context. Then we will propose a different threshold computation method. Wavelet packets can also be used to construct a regularization constraint, so we will study the efficiency of variational algorithms using such decompositions.

Many types of wavelets can be used to construct a packet basis ; they exhibit different properties depending on their spatial or frequential localization, and on their separability w.r.t. rows and columns. Decimated real wavelet transforms are efficient for satellite image deconvolution but produce artefacts since the transform is not translation invariant. To avoid these artefacts, the resulting image has to be averaged over all the possible integer translations, which slows down the algorithm.

1.2.4 Complex wavelet packet thresholding

There is a way to enable translation invariance without much loss of computational time, by using complex wavelets [33, 35]. Such wavelets also provide a better restoration by taking into account 6 directions. We have implemented a complex wavelet *packet* algorithm, which is detailed in chapter 3, where we also detail how to estimate the noise variance and the optimal threshold in each subband. The proposed method performs the inversion faster than shift invariant real transforms and reconstructs features of various orientations better.

Chapter 2

Real wavelet packet thresholding

2.1 Wavelet shrinkage

2.1.1 Wavelet denoising

We first deal with the case of denoising non-blurred images contaminated by white Gaussian noise.

The observed image is noisy and can be rewritten as $Y = \mathcal{X} + N$. We can decompose this image in an basis $\mathcal{B} = \{g_{k,m}\}$ which is an orthogonal basis of the discrete space $\mathrm{l}^2(\mathbb{R}^2)$, by using an orthogonal wavelet transform (see appendix A, section A.1). Here, k denotes the subband and m a spatial index within this subband. Thus, we have :

$$\langle Y, g_{k,m} \rangle = \langle \mathcal{X}, g_{k,m} \rangle + \langle N, g_{k,m} \rangle \tag{2.1}$$

We compute an estimator $\hat{Y}$ of $\mathcal{X}$ by applying an attenuation to each coefficient of the decomposition, i.e. by processing each coefficient separately :

$$\hat{Y} = \sum_{k,m} a_k(\langle Y, g_{k,m} \rangle)\, g_{k,m} \tag{2.2}$$

where a_k is called an attenuation function.

To enable an optimal denoising, by providing a maximum SNR between reconstructed and original images, an estimator $\hat{Y}$ must have a minimum risk $r(\hat{Y})$, defined by :

$$r(\hat{Y}) = E\left[\|\hat{Y} - \mathcal{X}\|^2\right] = E\left[\sum_m \Big(a\big(\langle \mathcal{X}, g_m \rangle + \langle N, g_m \rangle\big) - \langle \mathcal{X}, g_m \rangle\Big)^2\right] \tag{2.3}$$

where the expectation is taken w.r.t. the noise random variable. This risk depends on the function a. Donoho and Johnstone [16, 17, 18] have proven that taking a non-linear thresholding operator $a_k = \theta_T$ for the attenuation function a_k gives performances close to ideal coefficient selection (i.e. it provides a very small risk). This enables one to efficiently denoise the image Y, by keeping only the coefficients corresponding to the signal and strongly attenuating those corresponding to the noise. The thresholding function can be soft, hard, or more complicated. The choice of this function and of the associated threshold value T will be discussed later.

What properties should the basis $\mathcal{B}$ have? It has been proven in [17] that first, the image must have a sparse representation in $\mathcal{B}$, and second, the noise coefficients $\langle N, g_m \rangle$ must be nearly independent [16, 18] (which means that the covariance matrix of the noise should be *nearly diagonalized*).

For discrete images, such as satellite images contaminated by white Gaussian noise, the mirror wavelet basis [29, 31, 32] concentrates the original signal over a small number of coefficients (it is a well-known property, since wavelets have been widely used for image compression). The wavelet ψ has to provide sufficient frequency localization properties, by having a minimum of 4 vanishing moments [14]. Daubechies-8 or Symmlet-6 wavelets give good results for textures. More frequency-localized functions could be insufficiently space-localized and are not suited for sharp edge reconstruction.

The basis $\mathcal{B}$ is orthonormal, therefore the noise random variables N_{ij}, which are spatially independent, remain independent in the new basis. If W denotes a linear operator associated to a basis transform, the covariance matrix of the transform of a white independent noise of variance σ^2 is $\sigma^2 W W^t$. If W is orthonormal, we have $W W^t = I$: the covariance matrix of the noise is diagonal in the basis $\mathcal{B}$.

The signal is concentrated in the larger scale coefficients, while the noise is distributed with the same variance σ^2 over all coefficients. This means that signal and noise are located in nearly separate subbands, which makes possible an efficient thresholding estimation.

2.1.2 Variance of the deconvolved noise

In the case of deblurring, the deconvolved noise is no longer white, as explained hereafter. We are going to compute the variance of the deconvolved noise coefficients in a given basis $\mathcal{B}$ corresponding to one of the transforms which will be developed in the following sections.

First, let us compute the eigenvalues of the deconvolved noise N_{dec}. The deconvolution is achieved by dividing the Fourier transform of N by the Fourier transform of h in the frequency space :

$$\mathcal{F}[N_{dec}] = \frac{\mathcal{F}[N]}{\mathcal{F}[h]} \tag{2.4}$$

Since the Fourier basis is orthonormal, the transform coefficients of N remain independent and keep the same variance σ^2. Thus, the covariance matrix of N_{dec} is diagonalized by the Fourier transform. Then, its eigenvalues are simply given by :

$$(\sigma^2_{dec})_{ij} = E\left[|\mathcal{F}[N_{dec}]|^2\right] = \frac{\sigma^2}{|\mathcal{F}[h]|^2} \tag{2.5}$$

(see appendix C for a more detailed proof).

This clearly shows that the noise is colored by the deconvolution process. Since the convolution by h attenuates high frequencies, the inverse operation strongly amplifies the noise in these frequencies.

Since the wavelet transform is linear, the noise N_{dec} remains Gaussian. Its variance can be computed in the frequency space, because the wavelet transform is performed by circular

convolutions (as it will be explained in the following), corresponding to operators which are also diagonalized by a Fourier transform.

Let us consider an undecimated transform of the noise N_{dec} corresponding to subband k, which is achieved by a linear operator W^k, or a convolution with the kernel w^k. Then, in the same way as in equ. (2.5), the eigenvalues of the transform of N_{dec} are given by :

$$(\sigma_k^2)_{ij} = \sigma^2 \left| \frac{\mathcal{F}[W^k]_{ij}}{\mathcal{F}[h]_{ij}} \right|^2 \tag{2.6}$$

We use Parseval's theorem to compute the variance of the deconvolved noise in the wavelet transform domain, for a given subband. This variance only depends on the subband k (it is spatially constant), so it is only indexed by k. We finally obtain :

$$\sigma_k^2 = \sigma^2 \frac{1}{N_x N_y} \sum_{ij} \left| \frac{\mathcal{F}[W^k]_{ij}}{\mathcal{F}[h]_{ij}} \right|^2 \tag{2.7}$$

(see appendix C for a more detailed proof).

2.2 Necessity of using wavelet packets

Wavelet shrinkage has been used for image denoising, because the wavelet basis correctly adapts to the statistics of the white noise. We have seen that in the case of deconvolution, we have to deal with *colored* noise, and the previously described mirror wavelet basis no more allows the covariance matrix K to be nearly diagonal. As it is shown on Fig. 2.1, the high frequency subbands are contaminated by the deconvolved noise, so the signal present in these subbands is not recoverable by any thresholding method.

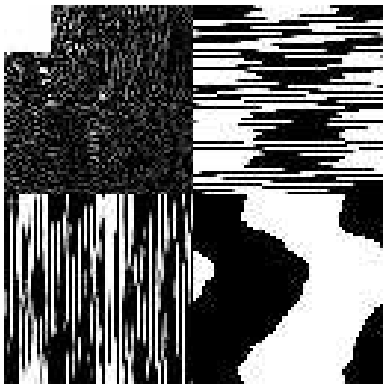


Figure 2.1: *Wavelet transform of the image of Fig. 1.1 deconvolved without regularization : the noise "explodes" in the highest frequencies subbands*

Kalifa and Mallat [29, 31] have shown that to construct a basis of approximate eigenvectors of K, wavelets must be replaced by *wavelet packets* [13]. To limit the growth of the

deconvolved noise variance at high frequencies, the finest scale subbands must be decomposed, in the same manner as the lowest scales in a classical wavelet transform. It avoids the "explosion" of the noise, since the wavelet packets have a frequential support which decreases exponentially at high frequencies.

Instead of splitting only the approximation at a given scale into details and a larger scale approximation, it is possible to *also decompose the details*, by using a quad-tree algorithm. It corresponds to splitting a detail space W_j into new spaces, deriving new bases.

Let us consider the generalization of the previously seen wavelet transform. We consider a quad-tree $\mathbf{Q}_J$ of maximum depth J. Let (p, q) denote a node of this tree at a given depth j. At each node of the tree, there is either a detail or an approximation subspace. In a wavelet tree, at each scale, only the node corresponding to an approximation subspace has descendants (see Fig. 2.2 a). In a wavelet packet tree, each node has descendants (4, if the decomposition is performed along rows, columns, or both rows and columns) (see Fig. 2.2 b). The subspace at node (p, q) is denoted $U_j^{p,q}$ and represents either an approximation, a wavelet detail, or a wavelet packet detail subspace.

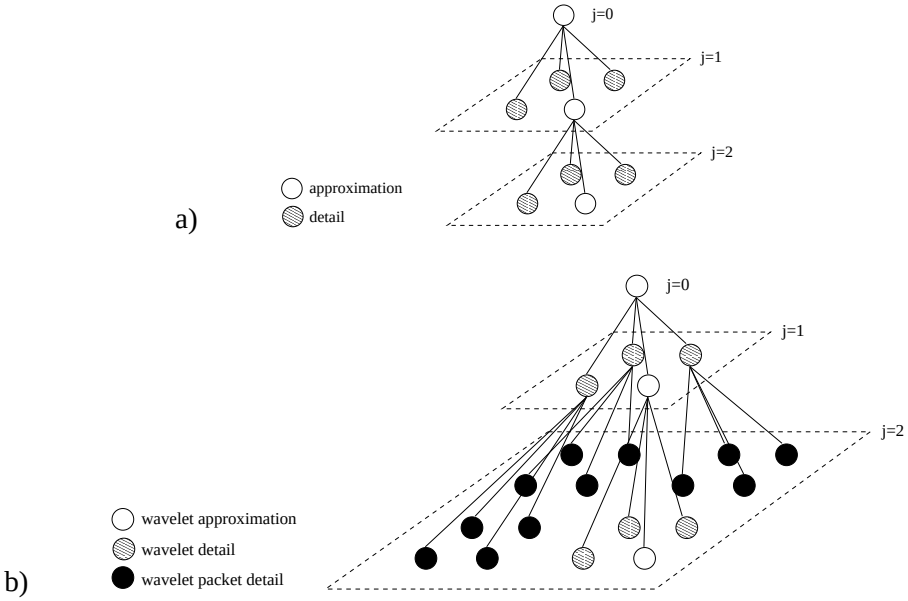


Figure 2.2: *a) wavelet tree, b) wavelet packet tree*

$\mathbf{Q}$ is said to be an admissible quad-tree, if each node at scale j either is a terminal node, or gives 4 new nodes at scale $j + 1$:

$$(j, p, q) \mapsto \begin{cases} \{(j{+}1, 2p, 2q), (j{+}1, 2p{+}1, 2q), (j{+}1, 2p, 2q{+}1), (j{+}1, 2p{+}1, 2q{+}1)\} \\ or \emptyset \quad \text{if} \quad (p, q) \neq (0, 0) \end{cases} \tag{2.8}$$

The bidimensional frequency space is divided into square regions of varying sizes. See Fig. 2.4 for an illustration.

The wavelet transform is generalized to a more general transform, based on functions denoted ψ^p and ψ^q, which include wavelets, scaling functions, as well as new functions called wavelet packets.

We can construct a wavelet packet basis of $\mathbf{L}^2(\mathbb{R}^2)$ [38]. An orthogonal basis of the wavelet packet separable space $U_j^{p,q}$ is given by :

$$\left\{\psi_{j,n}^p(x)\psi_{j,m}^q(y)\right\}_{(n,m)\in\mathbb{Z}^2} \tag{2.9}$$

To construct the bases associated with the wavelet packets, the scaling relations (A.4) and (A.7) (see appendix A) can be generalized to any function ψ_j^p, which generates the subspace U_j :

$$\psi_j^{2p}(t) = \sum_{-\infty}^{+\infty} h_n\, \psi_j^p(t - 2^j n) \tag{2.10}$$

$$\psi_j^{2p+1}(t) = \sum_{-\infty}^{+\infty} g_n\, \psi_j^p(t - 2^j n) \tag{2.11}$$

These functions are scaling functions like ϕ or wavelets like ψ, if ψ_j^p is a scaling function. If not, they are wavelet packets, since they are derived from a wavelet ψ at level j.

In the bidimensional case, the basis is separable, and we consider the following family, which defines an orthonormal basis of $\mathbf{L}^2(\mathbb{R}^2)$:

$$\left\{\psi_{j,n}^p(x)\psi_{j,m}^q(y)\right\}_{(n,m)\in\mathbb{Z}^2,\,(j,p,q)\in\mathbf{Q}} \tag{2.12}$$

A discrete image is therefore decomposed in the following orthonormal basis of $\mathrm{l}^2(\mathbb{R}^2)$, where $\mathbf{Q}_J$ is an admissible quad-tree of maximum depth J :

$$\left[\{\phi_{J,n}(k)\phi_{J,m}(l)\}_{(n,m)\in\mathbb{Z}^2}\,,\,\{\psi_{j,n}^p(k)\psi_{j,m}^q(l)\}_{(n,m)\in\mathbb{Z}^2,\,(j,p,q)\in\mathbf{Q}_J}\right]_{(k,l)\in\mathbb{Z}^2} \tag{2.13}$$

There are many possible choices for the quad-tree Q representing the wavelet packet transform. The idea is to segment the frequency plane into squares over which the noise variance σ_m^2 varies by a bounded factor from levels j and $j+1$. To achieve this, we only segment the highest frequency details, as illustrated in Fig. 2.4.

It is also possible to construct a hybrid tree, by using a mirror wavelet basis as in [47]. However, we did not notice a significant difference between the results computed with these different kinds of trees. Therefore we prefer to use the quadtree for sake of simplicity.

We can use a separable filter bank algorithm (see Fig. 2.3) to compute the discrete wavelet packet transform, as in the previous section. The previous algorithm is simply generalized to the detail images (the iteration to compute the wavelet transform remains the same).

$$\begin{array}{ll} d_{n,m}^{j+1,2p,2q} = (d^{j,p,q} \star \bar{h}\bar{h})_{2n,2m} & d_{n,m}^{j+1,2p+1,2q} = (d^{j,p,q} \star \bar{g}\bar{h})_{2n,2m} \\ d_{n,m}^{j+1,2p,2q+1} = (d^{j,p,q} \star \bar{h}\bar{g})_{2n,2m} & d_{n,m}^{j+1,2p+1,2q+1} = (d^{j,p,q} \star \bar{g}\bar{g})_{2n,2m} \end{array} \tag{2.14}$$

The reconstruction is performed in the following manner :

$$\begin{aligned} d_{n,m}^{j,p,q} = \; & (\check{d}^{j+1,2p,2q} \star hh)_{n,m} + (\check{d}^{j+1,2p+1,2q} \star gh)_{n,m} \\ & + (\check{d}^{j+1,2p,2q+1} \star hg)_{n,m} + (\check{d}^{j+1,2p+1,2q+1} \star gg)_{n,m} \end{aligned} \tag{2.15}$$

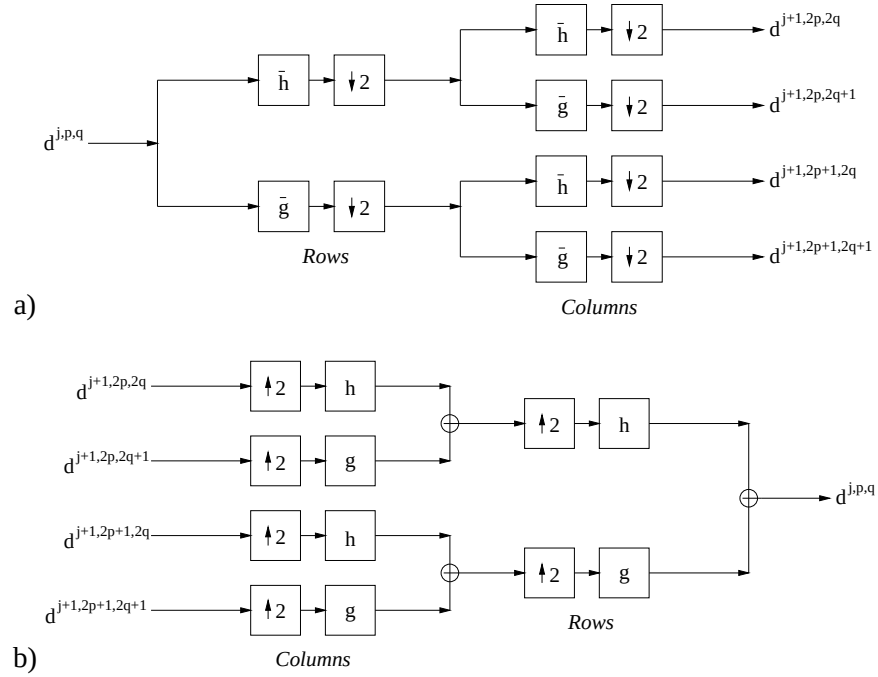


Figure 2.3: *Wavelet packet filter bank algorithms implemented with one-dimensional convolutions : a) decomposition, b) reconstruction*

On Fig. 2.5, it is clearly shown that some signal can be recovered even from the finest scale details, which was not the case for ordinary wavelets as in Fig. 2.1.

2.3 Optimal threshold estimation

2.3.1 Minimal risk of thresholding estimators

We have detailed a basis which is nearly optimal for noise filtering by thresholding the transform coefficients. We now have to determine which is the best choice for the threshold value associated with such an algorithm. There are several methods to compute the threshold T ; the first one we propose is based on the minimax risk calculus, and the second one tries to optimize the SNR by modeling the image transform subbands.

We denote X to be the image deconvolved without regularization. The variables x and ξ denote one coefficient of the wavelet packet transform of X and $\mathcal{X}$. If $g_{k,m}$ are the vectors

a)

a^3	d^{31}	d^{21}		d^{220}		d^{360}	d^{370}
d^{32}	d^{33}					d^{361}	d^{371}
d^{22}		d^{23}		d^{221}		...	...
						...	...
d^{202}		d^{212}		d^{222}		...	...
						...	...
...	...	...	...	...	...	...	...
...	...	...	...	...	...	...	...

b)

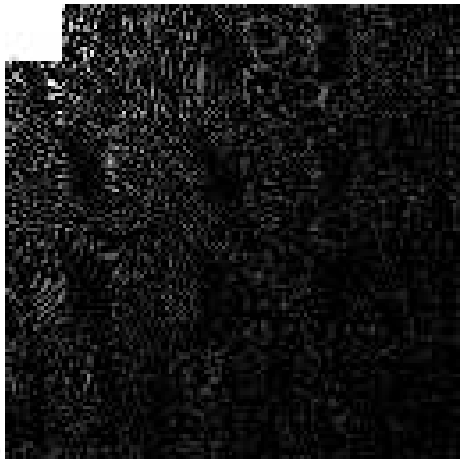


Figure 2.4: *a) rough partition of the frequency space introduced by the dyadic wavelet packet transform, b) wavelet packet transform of the original image of Fig. 1.1*

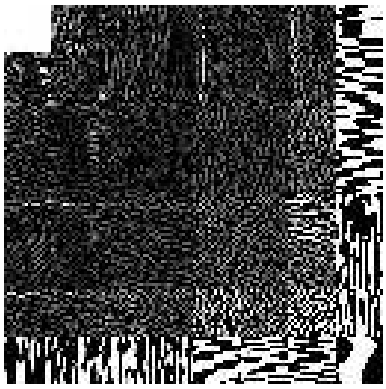


Figure 2.5: *Wavelet packet transform of the image of Fig. 1.1 deconvolved without regularization, showing the concentration of the noise in the high frequency subbands*

of the wavelet packet basis, where k is the index of the subband (for sake of simplicity)and m a spatial index, we have :

$$x = \langle X, g_{k,m} \rangle \,, \quad \xi = \langle \mathcal{X}, g_{k,m} \rangle \tag{2.16}$$

Since the noise N is white and Gaussian, equ. (1.1) gives :

$$x = \xi + n \qquad \text{where} \quad n = \mathcal{N}(0, \sigma_{k,m}^2) \tag{2.17}$$

and we consider the noise variance is constant in each subband, so we write $\sigma_{k,m} = \sigma_k$. We also assume that the noise coefficients are independent. These quantities can be computed by using equ. (2.7).

If we estimate the unknown coefficients by filtering the observations with an attenuation function a_k :

$$\hat{x} = a_k(x) \tag{2.18}$$

then the risk (2.3) of an estimator $\hat{X}$ can be rewritten as :

$$r(\hat{X}, \mathcal{X}) = E\left[\|\hat{X} - \mathcal{X}\|^2\right] = E\left[\sum_{k,m} \Big(a_k(x) - \xi\Big)^2\right] \tag{2.19}$$

The minimum risk is achieved by the following attenuation, called oracle attenuation [32] because in practice the original image $\mathcal{X}$ is not known :

$$a_k(x) = \frac{\xi^2}{\sigma_k^2 + \xi^2}\, x \tag{2.20}$$

(To derive this result, simply take the derivative of the risk (2.19) w.r.t. a_k and set it to zero. Notice that the independence of the noise coefficients is not necessary.)

An upper bound of the risk has been computed in [17], by choosing a soft threshold operator $a = \theta_T$. The classical soft and hard thresholding functions are defined by the following equations (see Fig. 2.6 for an illustration) :

$$\theta_T^s(x) = \begin{cases} x - T & \text{if } x > T \\ x + T & \text{if } x < -T \\ 0 & \text{if } |x| \leq T \end{cases} \qquad \theta_T^h(x) = \begin{cases} x & \text{if } |x| > T \\ 0 & \text{if } |x| \leq T \end{cases} \tag{2.21}$$

Donoho has defined the "universal threshold", equal to $T = \sigma_m \sqrt{2 \log_e N_x N_y}$. It guarantees that very little is present in the filtered coefficents. Furthermore, to improve the results, an infinite threshold is chosen for the subbands where σ_m is greater than the highest possible coefficient of the image transform. Indeed, the image is supposed to belong to a set Ω and to have bounded transform coefficients. Thus, choosing these thresholds provides a risk close to the risk of an oracle attenuation :

$$T_k = \begin{cases} \sigma_k \sqrt{2 \log_e N_x N_y} & \text{if } \sigma_k < \sup_{\mathcal{X} \in \Omega} |\xi| \\ \infty & \text{if } \sigma_k \geq \sup_{\mathcal{X} \in \Omega} |\xi| \end{cases} \tag{2.22}$$

In practice, Kalifa [29] has chosen $T_k \simeq 1.6\,\sigma_k$ for the finite threshold value. This value can be increased to restore images with fewer details and less noise, or decreased to reconstruct all the structures while accepting some noise. I can be considered as a parameter, and we will try to estimate it hereafter by optimizing the risk within a Bayesian framework.

A satisfying thresholding method could be constructed by using equ. (2.20) and trying to predict the unknown coefficients ξ. If we consider that a good prediction is achieved by $\xi \simeq a_k(x)$, then we have to solve :

$$a_k(x) = \frac{a_k(x)^2}{\sigma_k^2 + a_k(x)^2} x \tag{2.23}$$

There are multiple solutions. We retain the single solution which reduces to identity when there is no noise ($\forall k,\ \sigma_k = 0$). It gives the following function, which we call "oracle threshold" :

$$\theta_T^o(x) = \begin{cases} \frac{1}{2}\left(x + \sqrt{x^2 - T^2}\right) & \text{if } x > T \\ \frac{1}{2}\left(x - \sqrt{x^2 - T^2}\right) & \text{if } x < -T \\ 0 & \text{if } |x| \leq T \end{cases} \tag{2.24}$$

$$\text{with} \quad T = \begin{cases} 2\sigma_k & \text{if } \sigma_k < \sup_{\mathcal{X}\in\Omega}|\xi| \\ \infty & \text{if } \sigma_k \geq \sup_{\mathcal{X}\in\Omega}|\xi| \end{cases} \tag{2.25}$$

This expression naturally adapts to the variance of the noise in each subband. However, this approach, as for the other ones even if they adapt to the noise, does not take into account the statistics of the unknown image. These methods correspond to the least favourable prior case, i.e. they work with any type of input image. They also have a major drawback, consisting of removing too large a part of the useful signal.

We can wonder whether these methods are really optimal, or if we could find another one which would provide a smaller risk. By taking into account the prior distribution of the data, a more accurate thresholding could be possible. Therefore, determining a model of the image subbands is needed.

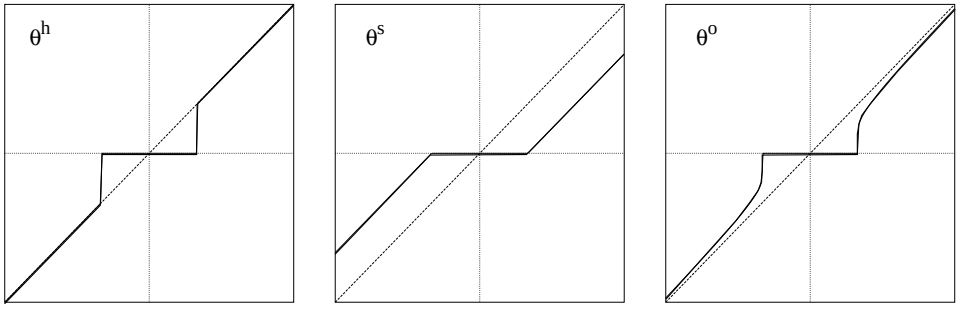


Figure 2.6: *Different thresholding functions : θ^h and θ^s (2.21), θ^o (2.24)*

2.3.2 Modeling the original image subbands

The coefficients of wavelet decompositions of satellite images can be represented by Generalized Gaussian distributions [1, 37, 43]. To be accurate, such models should be inhomogeneous. Nevertheless, it is possible to model each subband by a homogeneous Generalized Gaussian Random Field (GGRF), to construct a threshold estimation method. We have verified experimentally that the histogram of an original image subband can be simply modeled by :

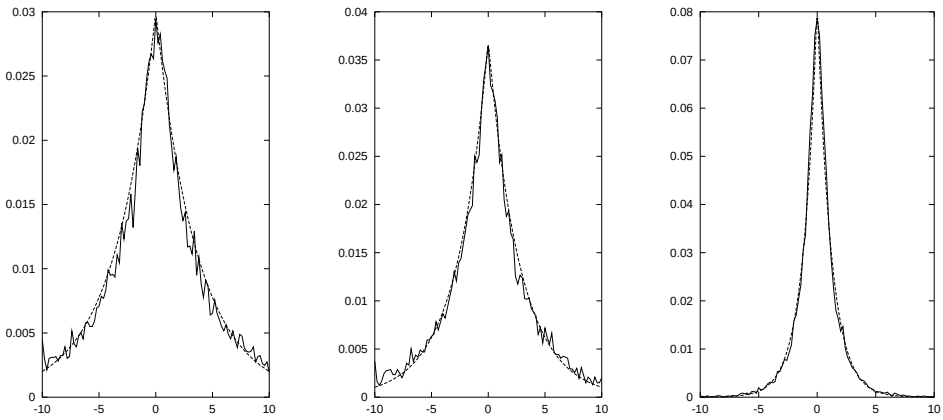
$$P(\xi) = \frac{p_k}{2\alpha_k \, \Gamma(1/p_k)} \, e^{-|\xi/\alpha_k|^{p_k}} \tag{2.26}$$

where α_k and p_k are the prior parameters of the model.

Moreover, the coefficients ξ can be considered to be independent in a given subband (even if it is not the case between different subbands).

As is shown by Fig. 2.7, the modeling remains sufficiently accurate if we set $p_k = 1$, in the case of satellite images :

$$P(\xi) = \frac{1}{2\alpha_k} \, e^{-|\xi|/\alpha_k} \tag{2.27}$$



(2,2,0) (2,2,1) (2,2,2)

Figure 2.7: *Distributions $P(\xi)$ of wavelet packet coefficients of the original image of Fig. 4.6 for subbands (j, p, q) ; the dashed curve corresponds to the prior model (2.27)*

On large size images, the prior parameter of a GGRF model can be efficiently computed by various methods, such as Maximum Likelihood (ML) for example. We choose to estimate the parameter α of the distribution from the histogram of a given subband, by simply computing the variance of the coefficients and comparing it to the noise variance.

The covariance matrix of the noise is nearly diagonal in a real wavelet packet basis [29]. We have verified experimentally that it is also true in the complex case (see Fig. 3.12), so we consider that the noise variables are independent in the complex wavelet packet basis.

The threshold selection can be made to minimize the risk for each subband k :

$$\hat{T}(\alpha_k, \sigma_k) = \arg\min_T \; E\left[(\theta_T(\xi + n) - \xi)^2\right] = \arg\min_T \; I(T, \alpha_k, \sigma_k) \tag{2.28}$$

where θ_T is one of the 3 thresholding functions defined by (2.21) and (2.24). It yields a nonlinear expression to be minimized, which can be achieved by a numerical method. Indeed, the function $I(T, \alpha_k, \sigma_k)$ is computed numerically :

$$I(T, \alpha_k, \sigma_k) = \frac{1}{2\alpha_k \sqrt{2\pi}\sigma_k} \iint_{(\xi,n)\in\mathbb{R}^2} (\theta_T(\xi + n) - \xi)^2 e^{-n^2/2\sigma_k^2 - |\xi|/\alpha_k} \, d\xi \, dn \tag{2.29}$$

The shape of this function is illustrated in Fig. 2.8 for different values of α_k and σ_k. Since it has a nearly quadratic behaviour near the optimum, the minimization w.r.t. T is made by a Newton-Raphson algorithm, where both the function I and its derivatives are numerically computed. The values of $\hat{T}$ computed this way can be stored in a table, to be used later within a thresholding algorithm. Figure 2.9 shows the variation of the optimal value $\hat{T}$ with α_k and σ_k. A good approximation of this value is given by the following equation :

$$\hat{T_k} \simeq 0.75 \frac{\sigma_k^2}{\alpha_k} \tag{2.30}$$

The experimental value $1.6\,\sigma_k$ found by Kalifa in [29] is justified, since the threshold given by the previous equation, for image of Fig. 1.1, is of the same order. The SNR of the reconstructed image is improved by taking a more accurate value, as defined by (2.30), since the SNR is maximal for a minimal risk $r(\hat{X}, \mathcal{X})$.

To compute $\hat{T}$ the value of α_k is needed. It can be estimated from a noisy subband, by computing the variances of each distribution. We have $x = \xi + n$ for each coefficient. We assume that signal ξ and noise n are independent : $E[x^2] = E[\xi^2] + E[n^2]$. The expression of the distribution (2.26) gives $E[\xi^2] = 2\alpha_k^2$.

$E[x^2]$ can be estimated on the subband k by a sum of squares, n_k is the number of coefficients of this subband. So we obtain :

$$\alpha_k = \sqrt{\frac{\frac{1}{n_k}\sum_i x_i^2 - \sigma_k^2}{2}} \tag{2.31}$$

2.3.3 The Bayesian approach

It is possible to estimate the unknown wavelet coefficients within a Bayesian framework, as introduced in [43]. We have defined a prior distribution for these coefficients. To compute the MAP estimate of ξ, we use Bayes theorem to calculate the expression of the posterior probability :

$$\hat{\xi} = \arg\max_\xi P(\xi \mid x) = \arg\max_\xi P(x \mid \xi) P(\xi) \tag{2.32}$$

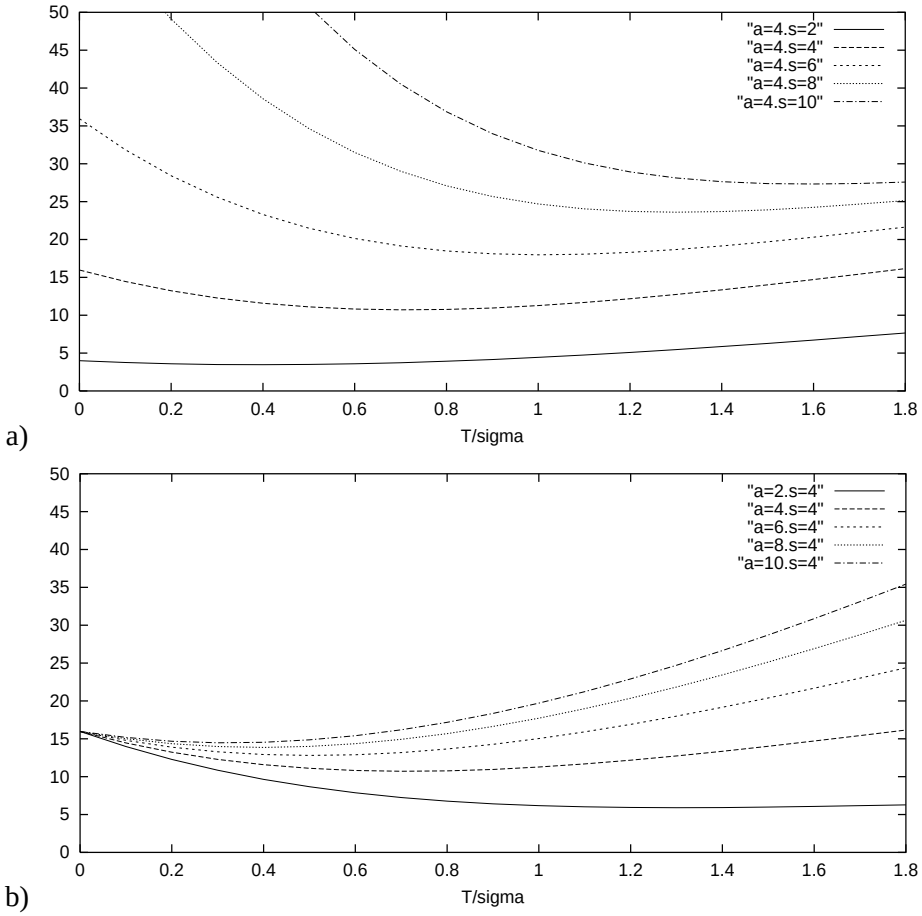


Figure 2.8: *a) Risk as a function of T/σ_k computed for different values of σ_k ("s") with $\alpha = 4$, b) Risk computed for different values of α ("a") with $\sigma_k = 4$*

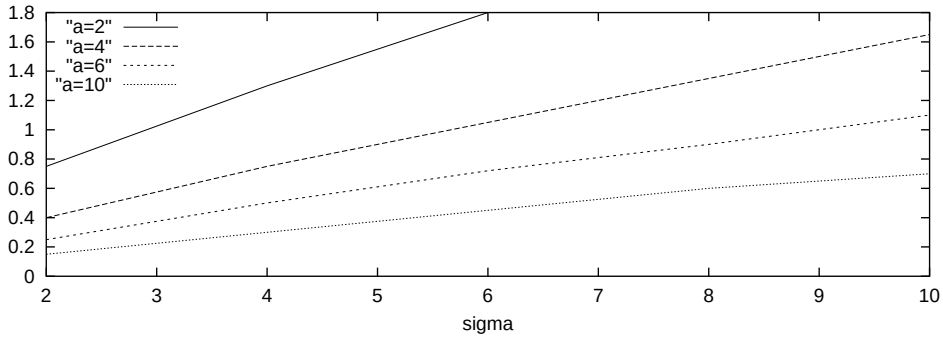


Figure 2.9: *$\hat{T}/\sigma_k$ as a function of σ_k, computed for different values of α ("a")*

where $P(x \mid \xi)$ is given by the distribution of the noise, since $x = \xi + n$:

$$P(x \mid \xi) = \frac{1}{\sqrt{2\pi}\sigma_k} e^{-(x-\xi)^2/2\sigma_k^2} \tag{2.33}$$

Then we obtain, by using the prior law (2.26), the following expression of the MAP :

$$\hat{\xi} = \arg\min_{\xi} \left[(x-\xi)^2/2\sigma_k^2 + |\xi|/\alpha_k \right] \tag{2.34}$$

It is possible to show [43] that it is equivalent to apply a soft thresholding operator θ_T to each coefficient, and the threshold T is directly given by expression (2.34) :

$$\hat{\xi} = \theta^s_{\hat{T}}(x) \quad \text{with} \quad \hat{T} = \frac{\sigma_k^2}{\alpha_k} \tag{2.35}$$

Thus, the soft thresholding method presented before naturally comes from the Bayesian approach, if we suppose the unknown variables to follow a Laplacian prior law (2.27). Furthermore, the threshold $\hat{T}$ is known, and it is given by σ_k^2/α_k. This expression is close to the previously found law (2.30).

It is also possible to generalize the model (2.26) by choosing a Generalized Gaussian with an exponent p not necessary equal to 1. The resulting thresholding functions are smoother for a higher p, and they become linear if $p = 2$.

We did not notice a significant improvement of the results by modeling the unknown coefficients with $p \neq 1$. Our experiments have shown that the Bayesian approach is preferable to the minimum risk computation. So we will use equ. (2.35) instead of (2.30) to estimate the optimal threshold in the final deconvolution algorithm.

2.3.4 Noise variance estimation

Let us recall equ. (2.7), which gives an estimate of the variance of the deconvolved noise in each subband k. This expression depends on the operators W^k, which are used to perform a non-decimated wavelet packet decomposition (the decimation does not modify the variances, but complicates the computations, that is why we consider non-decimated transforms here).

The operators W^k have to be expressed using the filters h and g. $|\mathcal{F}[g]|^2$ is computed from $|\mathcal{F}[h]|^2$ using the mirror filter property (A.8) (see appendix A). The Fourier transforms of the filters at different scales j are denoted $\mathcal{F}[h]^j$. The kernel w^k is separable. Table 2.1 gives the expression of w^k and σ_k/σ for the used subbands according to the labeling of Fig. 2.4.

2.3.5 The wavelet packet thresholding algorithm

The deconvolution method in [29], completed by an automatic threshold selection as seen in section 2.3.3, consists of the following steps :

k	Subband	w^k	σ_k/σ
1	d^{21}	$\bar{h}_x * \bar{h}_y * \bar{g}_x^2 * \bar{h}_y^2$	3.3
2	d^{22}	$\bar{h}_x * \bar{h}_y * \bar{h}_x^2 * \bar{g}_y^2$	3.2
3	d^{23}	$\bar{h}_x * \bar{h}_y * \bar{g}_x^2 * \bar{g}_y^2$	6.9
4	$d^{2\,20}$	$\bar{g}_x * \bar{h}_y * g_x^2 * g_y^2$	9.2
5	$d^{2\,21}$	$\bar{g}_x * \bar{h}_y * g_x^2 * h_y^2$	19
6	$d^{2\,02}$	$\bar{h}_x * \bar{g}_y * g_x^2 * g_y^2$	8.0
7	$d^{2\,12}$	$\bar{h}_x * \bar{g}_y * h_x^2 * g_y^2$	17

Table 2.1: *Expression of the filters corresponding to different subbands, with a Symmlet-6 filter, for the wavelet packet transform of Fig. 2.4, the input image and the kernel h are given by Fig. 1.1*

ALGORITHM 2.3.1 (WAVELET PACKET THRESHOLDING)

- *DCT (discrete cosine transform) of the observation Y*
- *Deconvolution : divide by $\mathcal{F}[h]$ (assuming nonnull coefficients)*
- *Inverse DCT of the result, which gives X*
- *WPT (wavelet packet transform) of X*
- *Computation of σ_k using the known h and σ*
- *Estimation of α_k from WPT[X] and σ_k, using equ. (2.31)*
- *Coefficient thresholding, using (2.35)*
- *Inverse WPT, which gives the estimate $\hat{X}$.*

To avoid oscillations near the borders of the images, a DCT replaces the FFT without an increase of computational complexity [25].

In practice, for the image provided by the French Space Agency (CNES), only 7 subbands are needed for the filtering algorithm. The highest frequencies are set to 0 because the highest coefficient in these bands is lower than σ_k, which means that these subbands only contain noise. These values can be used for any type of SPOT 5 simulated image. The values of α_k are not given since they depend on the image.

Fig. 2.11 shows the result of the thresholding method presented in this chapter. The results for the whole image are presented in chapter 4. To improve the SNR, the subbands of the wavelet transform d^{11}, d^{12}, d^{13} of the result could be filtered by hard thresholding, with $T_k = 3\sigma_k^r$, where σ_k^r is the standard deviation of the residual noise [29].

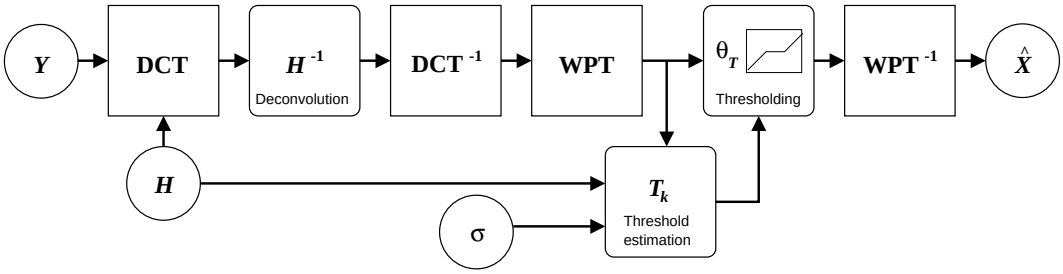


Figure 2.10: *Automatic thresholding algorithm : deconvolution in the Cosine transform basis (DCT), threshold estimation and thresholding in the wavelet packet basis (WPT)*

a) b)

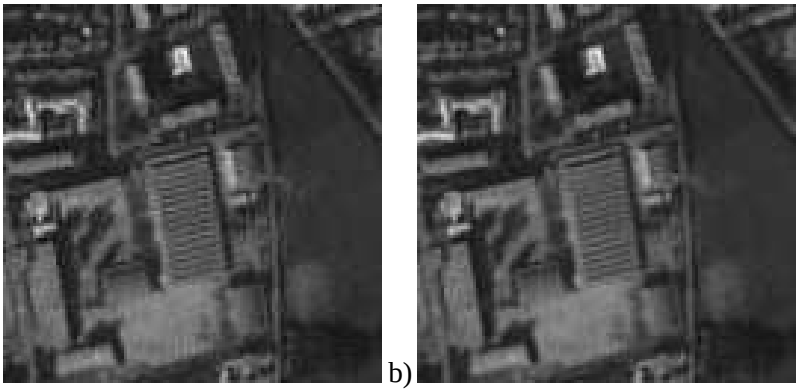


Figure 2.11: *Results of the wavelet packet thresholding algorithm on image of Fig. 1.1 : a) without averaging, b) averaged over 16 translations*

2.4 Drawbacks of real wavelet transforms

2.4.1 Lack of shift invariance

As seen on Fig. 2.11, thresholding produces artefacts, because the wavelet decomposition is not translation invariant, and the image exhibits discontinuities, like edges or sharp textures. Therefore, the estimator $\hat{X}$ has to be averaged over all possible translations corresponding to the chosen decomposition level. In the case of the image of Fig. 2.11, 16 translations are sufficient, as we stop at scale $j = 2$. To speed up the filtering, 4 to 8 translations could be sufficient but some ripple effects can remain.

The shift invariance is achieved in the following manner [12] :

$$\hat{X}_{ij}^{inv} = \frac{1}{u_m v_m} \sum_{u=0,v=0}^{u_m,v_m} \hat{X}_{i+u,j+v} \tag{2.36}$$

where m represents the maximum shift. This obviously slows down the thresholding algorithm by a factor 4 to 16.

Real shift invariant wavelet transforms exist, but they are highly redundant, and the redundancy grows with the depth of the transform. The reconstruction is done by averaging, since there is not an unique reconstruction. It provides good quality results, but slows down the algorithm, especially when the depth of the transform is high.

A Laplacian pyramid can be constructed [38]. It is possible to use the “à trous” algorithm to generate a perfect reconstruction shift invariant transform ; this type of method has been applied to astronomical image deconvolution [6]. However, it does not provide satisfying results essentially because of the isotropy of the lowpass filter, which is not compatible with the small oriented details present in satellite images.

It is also possible to process an overcomplete expansion of the image, as non-decimated transforms (in a basis which is not orthogonal), as we will explain in section 2.5, but the performance of such algorithms is not sufficient, regarding both image quality and computing time.

2.4.2 Poor directionality

Thanks to their separability, the wavelet packets can be computed by fast filter bank algorithms. But the consequence of this separability is to divide the frequency space into subbands oriented along horizontal or vertical directions. Other directions, such as the diagonals, are not correctly represented by such a decomposition. This is visible on variously oriented features, such as roads or oriented textures, which generally exhibit aliasing artefacts, and could disappear in some cases after thresholding because they cannot be separated from the noise.

Non-separable real decompositions could be used, but the gain of directionality would be at the cost a significant increase in computational complexity.

2.5 Comparison with wavelet regularization

2.5.1 Wavelet packet regularization models

It is possible to regularize the ill-posed problem of the deconvolution of chapter 1 by introducing a wavelet-based regularizing function. A few authors have used such an approach to solve image restoration problems [4, 36, 39, 40].

This a priori constraint can be chosen to be translation invariant, to avoid artefacts coming from the decimation. Undecimated transforms can be introduced in the regularizing function, by means of separable convolutions with filters h and g at different scales, in the same manner as in the previous section. To enable filtering at different scales while keeping the same size of the image, the "à trous" algorithm has to be used [38].

We can take the previous wavelet packet approach, which has been successfully used to construct a thresholding method. We have developed a variational algorithm working with the same basis $\mathcal{B}$. We have to impose a constraint on the highest frequency subbands. The idea is to penalize the noise in each subband, while preserving the signal, in the same manner that pixel differences are penalized in [10, 20, 21] by a φ-model. The function φ is called a potential function (see Fig. 2.12 for an illustration). Properties of the φ-functions have been studied in a variational approach in order to preserve the edges, avoiding noise amplification [3].

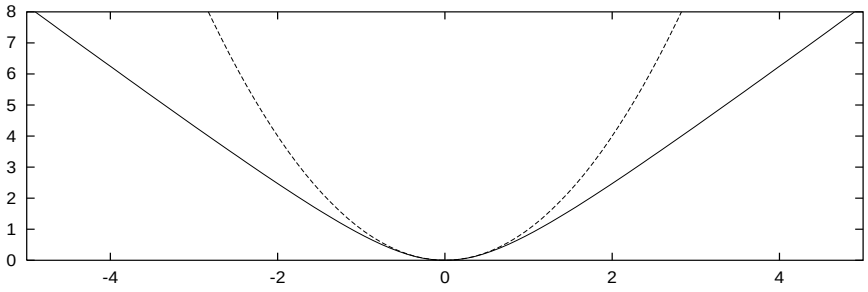


Figure 2.12: *Solid : Hyper Surface φ-function $\varphi(u) = 2\sqrt{1+u^2} - 2$, dashed : quadratic function $\varphi(u) = u^2$*

The solution is the image X which minimizes the following criterion, which includes a data term $\|Y - h \star X\|^2$ and a regularizing term involving the highest frequency subbands :

$$\hat{X} = \arg\min_X \left[\|Y - h \star X\|^2 / 2\sigma^2 + \sum_k \lambda_k \sum_{i,j} \varphi\left(\frac{(W^k X)_{ij}}{\delta_k} \right) \right] \tag{2.37}$$

where λ_k and δ_k are the regularizing hyperparameters associated to the subband k. This extends the nearest neighbour φ-model to a more general approach, in which longer distance interactions are allowed between the pixels.

It is possible to define a very simple operator W corresponding to the Haar transform, which is the simplest wavelet transform, with $h = (1/\sqrt{2}, 1/\sqrt{2})$. This is equivalent to using a classical φ-model, where the difference operators are averaged w.r.t. the orthogonal direction, i.e. the derivative is computed after convolving the image with the smoothing kernel h. The models are equivalent only if the first scale of the transform is considered. Results can be improved by adding other scales, enabling wavelet packet regularization.

The Haar transform is not convenient for texture preservation, therefore a wavelet with 4 or more vanishing moments has to be used to enable a better frequency localization. Symmlet-6 wavelets can be used as in the previous thresholding algorithm. Furthermore, the highest frequency subbands should be highly penalized since they only contain noise. Thus, it would be possible to recover the textures better than with a classical φ-model based on first order derivatives.

To minimize the regularized criterion (2.37) it is possible to use a half-quadratic extension of φ as in [9, 10], by adding auxiliary images $\{B^k\}$. Alternate minimizations w.r.t X and $\{B^k\}$ are used to compute the solution. The minimization w.r.t. X is achieved by diagonalizing the quadratic form Σ as in [27, 25] by a Fourier transform, where Σ is given by :

$$\Sigma = H^t H/2\sigma^2 + \sum_k \frac{\lambda_k}{\delta_k}(W^k)^t(W^k)$$

where H is the block circulant matrix corresponding to the convolution by h. To avoid oscillations near the borders of the images, a DCT can replace the FFT without an increase of computational complexity. Both minimizations are condensed in a single iteration in the following algorithm. The results are presented in Fig. 2.13.

The minimization w.r.t. the auxiliary variables is achieved by using the half-quadratic theorem [3], which involves the function φ_p defined by $\varphi_p(u) = u - \varphi'(u)/2$. Here we denote D the diagonal form of Σ. Then we use the following algorithm :

ALGORITHM 2.5.1 (WAVELET PACKET REGULARIZATION)

- *Initialization :* $X^0 = Y$ *;*

$$D = \frac{|\mathcal{F}[h]|^2}{2\sigma^2} + \sum_k \frac{\lambda_k}{\delta_k}|\mathcal{F}[w^k]|^2$$

- *Repeat the following step, until convergence :*

$$X^{n+1} = \mathcal{F}^{-1}\left\{D^{-1}.\left(|\mathcal{F}[h]|^2\mathcal{F}[Y] + \mathcal{F}\left[\sum_k \frac{\lambda_k}{\delta_k}(W^k)^t\ \varphi_p\left(\frac{W^k X^n}{\delta_k}\right)\right]\right)\right\}$$

2.5.2 Problems raised by such models

The wavelet packet based regularizing model (2.37) does not provide the same quality results as the corresponding thresholding algorithm.

First, the computing time is much longer because of the iterative nature of this type of algorithm. Indeed, 10 or more iterations are necessary to reach convergence, and each one

Figure 2.13: *Results of Algo. 2.5.1 on the image of Fig. 1.1 (Daubechies-8 wavelet)*

needs the computation of all undecimated wavelet packet transforms corresponding to the different subbands, which cost more than a decimated wavelet transform.

Second, the resulting image is not sufficiently denoised, since the highest frequencies are not set to 0 as in thresholding methods, but only penalized. The parameters of the highest frequency subbands have to be chosen as high as possible to filter the noise, but sufficiently small to avoid the instabilities which appear sometimes on the borders, due to approximate boundary conditions, and to the iterative nature of the algorithm.

A difficulty of wavelet regularizing models is the estimation of the parameters. There are 2 parameters for each subband with the model defined by (2.37). Some methods to estimate them can be found in the literature [4]. We could also apply the methods detailed in [28]. Finally, the results are not sufficiently good to justify such a complexity.

As a conclusion, wavelet packet shrinkage methods seem to be more suitable for image deconvolution than these variational regularizing models. The wavelet packet basis, well adapted to thresholding techniques, seems not to provide satisfactory results within a variational context. The shift variance problem can be solved by a great increase of computational complexity, but the denoising process cannot be completely achieved by such methods.

Chapter 3

Complex wavelet packet thresholding

In this chapter, we will first present a particular wavelet packet transform, which provides useful invariance and directional properties and which can be interpreted as a complex transform. Then, a new thresholding algorithm will be detailed, based on this decomposition, and derived from the one presented in the previous chapter.

3.1 Complex wavelets

To understand the complex wavelet decompostion, we first have to recall the biorthogonal wavelet transform (see appendix A, section A.2).

3.1.1 Unidimensional complex wavelet transform

Kingsbury has introduced in [33, 34] a complex wavelet transform which enables perfect reconstruction (PR). This property is essential if we want to develop a thresholding algorithm like the one proposed by Kalifa in [31], detailed in the previous chapter.

Biorthogonal wavelet transforms provide PR, and Kingsbury shows that it is possible to use them in a particular manner to achieve a PR nearly *shift invariant* complex wavelet transform. He has developed a dual-tree algorithm, by noting that an approximate shift invariance can be obtained with a real biorthogonal transform by doubling the sampling rate at each scale. This is achieved by computing two parallel wavelet trees, which are subsampled differently.

The shift invariance is perfect at level 1, since the transform is simply a non-decimated transform, whose coefficients are re-ordered into 2 interleaved trees, taking into account their parity. This property is only approximately achieved beyond level 1, and the proposed algorithm is constructed in such a way to optimize the translation invariance.

The approximations and details for the two trees are denoted respectively (a_A, d_A) and (a_B, d_B). The following algorithm shows how to perform the transform, and it is illustrated by Fig. 3.1.

ALGORITHM 3.1.1 (1D COMPLEX WAVELET TRANSFORM)

- *At level 1, there is one sample offset between the trees. It is sufficient to keep odd samples in one case and even samples in the other, i.e. we have :*

$$\begin{aligned} (a_A^1)_n &= (a^0 \star h^o)_{2n} & (d_A^1)_n &= (a^0 \star g^o)_{2n} \\ (a_B^1)_n &= (a^0 \star h^o)_{2n+1} & (d_B^1)_n &= (a^0 \star g^o)_{2n+1} \end{aligned} \tag{3.1}$$

- *At level $j > 1$, there must be half a sample difference between the trees. This is only possible if the filters have a different length for each tree, i.e. it requires even-length filters h^e, g^e in one tree and odd-length filters h^o, g^o in the other :*

$$\begin{aligned} (a_A^{j+1})_n &= (a_A^j \star h^e)_{2n} & (d_A^{j+1})_n &= (a_A^j \star g^e)_{2n} \\ (a_B^{j+1})_n &= (a_B^j \star h^o)_{2n+1} & (d_B^{j+1})_n &= (a_B^j \star g^o)_{2n+1} \end{aligned} \tag{3.2}$$

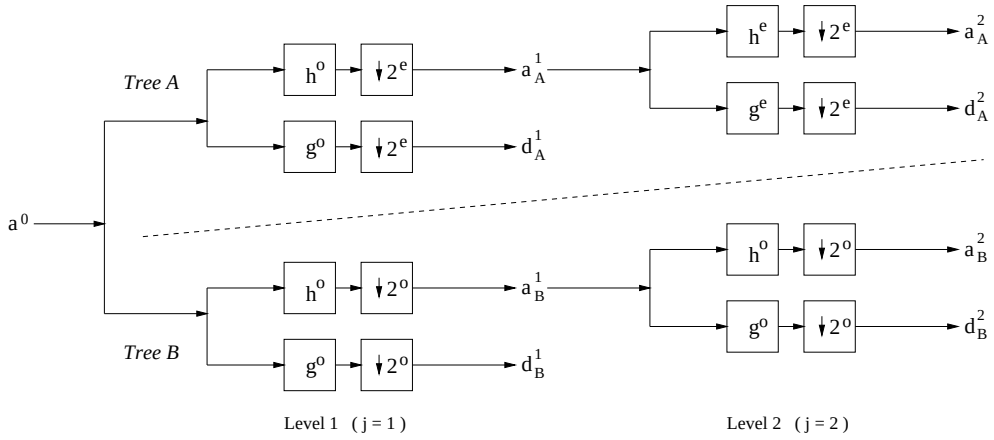


Figure 3.1: *Dual tree filter bank for the unidimensional complex wavelet transform*

The algorithm needs two pairs of filters h^o, g^o and h^e, g^e. As explained in [34, 35], the shift invariance property is approximately achieved beyond level 1 by this combination of odd and even length filters. Thus, using different parity filters for each tree provides half a sample shift between the trees. When filtering a signal by keeping only the coefficients of a given level j, it is shown that the aliasing is much smaller when the filters have different parity.

The filters are the same in the two trees for $j = 1$ (odd) and different for $j > 1$, hence in one tree the filter is not the same between the first and second scale. Therefore, the corresponding wavelet changes between the two first scales, which is surprising compared to the classical multiresolution analysis, where the same mother wavelet ψ is translated and dilated to generate an orthogonal basis of $\mathbf{L}^2(\mathbb{R})$.

In practice, we use near-symmetric (13,19) tap filters for h^o and $\tilde{h}^o$, designed with a transformation of variables method [34]. The (12,16) tap even filters h^e and $\tilde{h}^e$ are constructed to match as closely as possible (in the mean-square sense) to the odd ones. The

corresponding g filters have respectively odd and even symmetry, which is also the case for the corresponding wavelets.

Perfect reconstruction filter banks can be implemented if the conditions (A.28) and (A.26) (see appendix A) are verified at each scale.

For the second tree B, the filters are all the same, then a function is decomposed in a biorthogonal basis (A.21), and a discrete signal in a discrete basis (A.29),(A.30) (see appendix B).

For the first tree A, the transform consists of decomposing the signal in a discrete basis which mixes the two different types of wavelets, odd and even (see appendix B).

The details d_A and d_B can be interpreted as the real and imaginary parts of a complex process $z = d_A + i\, d_B$. The essential property of this transform is that the magnitude of the step response is approximately invariant with the input shift, while only the phase varies rapidely (see [35] for a good illustration). If we only consider the magnitude $|z|$ for a given scale, it corresponds to an approximately shift invariant transform, and thresholding this magnitude produces less artefacts than thresholding real transforms.

It is not really a complex wavelet transform, since it does not use any complex wavelet. It is implemented with real wavelets. (It is not really a wavelet transform because of the variety of filters.) Classical complex transforms (using complex-valued wavelets) cannot at the same time give useful frequency responses and allow perfect reconstruction.

The reconstruction is done in each tree independently, by using the dual filters, and the results are averaged to obtain a^0 to ensure the symmetry between the trees, thus producing the desired shift invariance. This is illustrated by Fig. 3.2.

ALGORITHM 3.1.2 (1D INVERSE COMPLEX WAVELET TRANSFORM)

- *For each scale $j > 0$:*

$$\begin{aligned}(a_A^j)_n &= (\check{a}_A^{j+1} \star \tilde{h}^e)_n + (\check{d}_A^{j+1} \star \tilde{g}^e)_n \\ (a_B^j)_n &= (\hat{a}_B^{j+1} \star \tilde{h}^o)_n + (\hat{d}_B^{j+1} \star \tilde{g}^o)_n \end{aligned} \tag{3.3}$$

- *For $j = 0$, the trees are averaged :*

$$a_n^0 = \frac{1}{2}\Big((\check{a}_A^1 \star \tilde{h}^o)_n + (\check{d}_A^1 \star \tilde{g}^o)_n + (\hat{a}_B^1 \star \tilde{h}^o)_n + (\hat{d}_B^1 \star \tilde{g}^o)_n\Big)$$

$$\textit{where} \quad \check{x}_n = \begin{cases} x_p & \textit{if} \quad n = 2p \\ 0 & \textit{if} \quad n = 2p+1 \end{cases} \quad , \quad \hat{x}_n = \begin{cases} x_p & \textit{if} \quad n = 2p+1 \\ 0 & \textit{if} \quad n = 2p \end{cases} \tag{3.4}$$

3.1.2 Bidimensional complex wavelet transform

The complex wavelet transform filter bank

The previous approach can be generalized to the bidimensional case, by using a separable filter bank technique, as seen in section A.1.2 for real wavelets. Kingsbury [34] has

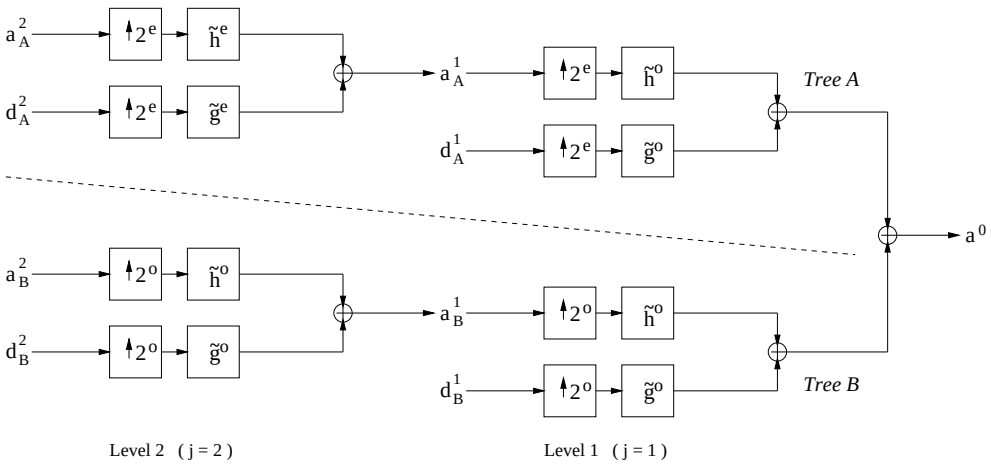


Figure 3.2: *Dual-tree filter bank for the unidimensional inverse complex wavelet transform*

shown that such a transform remains computationally efficient, and provides better directional properties than real transforms, in spite of its separability. The PR property is conserved, and enables one to set up thresholding methods like in section 2.1.

The shift invariance is perfect at level 1. Indeed, this is simply a non-decimated transform, whose coefficients are re-ordered into 4 interleaved trees A, B, C, D, taking into account their parity w.r.t. lines and columns (see Fig. 3.3) [35]. This property is only approximately achieved beyond level 1 (and the proposed combination of filters is optimal in the sense of shift invariance, as in the unidimensional case).

The approximations and details of the 4 trees are denoted respectively (a_A, d_A) to (a_D, d_D). The following algorithm shows how to perform the quad-tree transform, and it is illustrated by Fig. 3.4. It is equivalent to decompose the image in a discrete bidimensional complex wavelet basis (detailed in appendix B).

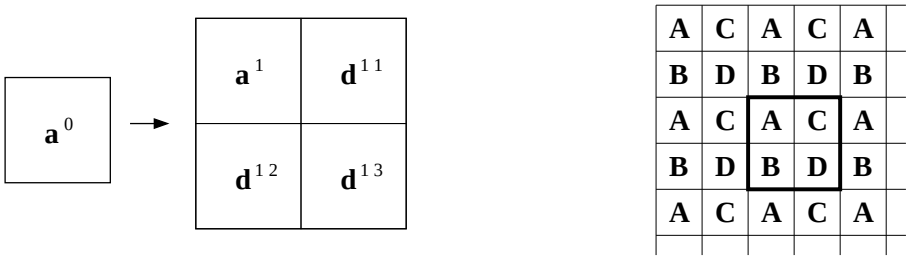


Figure 3.3: *First level of the bidimensional complex transform. Left : non-decimated transform (4:1 redundancy), right : interleaving of the coefficients corresponding to the 4 trees in the non-decimated transform*

ALGORITHM 3.1.3 (2D COMPLEX WAVELET TRANSFORM)

- *At level 1, there is one sample offset between the 4 trees. Compute a non-decimated wavelet transform with h^o and g^o, then separate odd/even coeffi cients (indices o and e) to extract the trees (see Fig. 3.3):*

Tree T	**A**	**B**	**C**	**D**
$(a_T^1)_n$	$(a^0 \star h^o h^o)_{e,e}$	$(a^0 \star h^o h^o)_{e,o}$	$(a^0 \star h^o h^o)_{o,e}$	$(a^0 \star h^o h^o)_{o,o}$
$(d_T^{1,1})_n$	$(a^0 \star g^o h^o)_{e,e}$	$(a^0 \star g^o h^o)_{e,o}$	$(a^0 \star g^o h^o)_{o,e}$	$(a^0 \star g^o h^o)_{o,o}$
$(d_T^{1,2})_n$	$(a^0 \star h^o g^o)_{e,e}$	$(a^0 \star h^o g^o)_{e,o}$	$(a^0 \star h^o g^o)_{o,e}$	$(a^0 \star h^o g^o)_{o,o}$
$(d_T^{1,3})_n$	$(a^0 \star g^o g^o)_{e,e}$	$(a^0 \star g^o g^o)_{e,o}$	$(a^0 \star g^o g^o)_{o,e}$	$(a^0 \star g^o g^o)_{o,o}$

- *At level $j > 1$, there must be half a sample difference between the trees. This is only possible if the fi lters have a different length for each tree, i.e. it requires even-length fi lters l^e, g^e in one tree and odd-length fi lters l^o, g^o in the other:*

Tree T	**A**	**B**	**C**	**D**
$(a_T^{j+1})_n$	$(a_A^j \star h^e h^e)_{e,e}$	$(a_B^j \star h^e h^o)_{e,o}$	$(a_C^j \star h^o h^e)_{o,e}$	$(a_D^j \star h^o h^o)_{o,o}$
$(d_T^{j+1,1})_n$	$(a_A^j \star g^e h^e)_{e,e}$	$(a_B^j \star g^e h^o)_{e,o}$	$(a_C^j \star g^o h^e)_{o,e}$	$(a_D^j \star g^o h^o)_{o,o}$
$(d_T^{j+1,2})_n$	$(a_A^j \star h^e g^e)_{e,e}$	$(a_B^j \star h^e g^o)_{e,o}$	$(a_C^j \star h^o g^e)_{o,e}$	$(a_D^j \star h^o g^o)_{o,o}$
$(d_T^{j+1,3})_n$	$(a_A^j \star g^e g^e)_{e,e}$	$(a_B^j \star g^e g^o)_{e,o}$	$(a_C^j \star g^o g^e)_{o,e}$	$(a_D^j \star g^o g^o)_{o,o}$

Complex interpretation of the quad-tree transform

For each subband, the details d_A, d_B, d_C, d_D can be interpreted as the real and imaginary parts of two complex processes z_+ and z_- :

$$\begin{aligned} z_+^{j,k} &= (d_A^{j\,k} - d_D^{j\,k}) + i\,(d_B^{j\,k} + d_C^{j\,k}) \\ z_-^{j,k} &= (d_A^{j\,k} + d_D^{j\,k}) + i\,(d_B^{j\,k} - d_C^{j\,k}) \end{aligned} \tag{3.5}$$

Thus, the details of the 3 subbands corresponding to hg, gh and hh provide 6 complex subbands, instead of 3 real subbands in the real case. The frequency domain interpretation of this decomposition, which is one of the essential points, will be discussed in section 3.1.3.

As in the unidimensional case, the magnitude of the step response is approximately invariant with the input shift, while the phase varies linearly. If we only consider the magnitudes $|z_\pm|$ for a given scale, it corresponds to an approximately shift invariant transform, and thresholding this magnitude produces less artefacts than thresholding real transforms.

We again have to notice that it is not really a complex wavelet transform, since it does not use any complex wavelet function. Nevertheless, the quad-tree transform has in practice the same properties as a complex transform w.r.t. shifting of the input image.

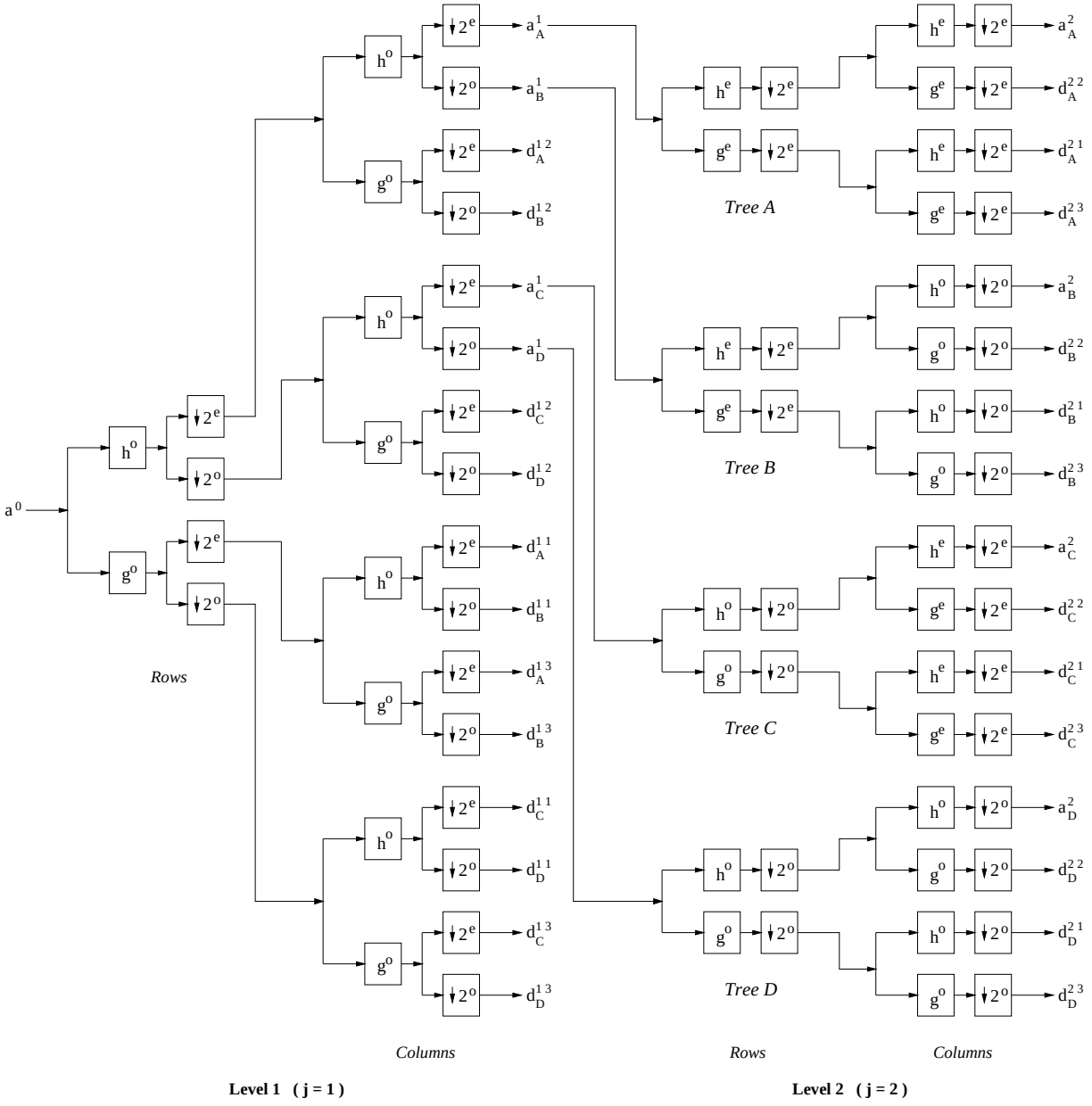


Figure 3.4: *Quad-tree filter bank for the bidimensional complex wavelet transform*

Inverse complex wavelet transform

The reconstruction is done in each tree independently, by using the dual filters, and the results of the 4 trees are averaged to obtain a^0 to ensure the symmetry between the trees, thus enabling the desired shift invariance. This is illustrated by Fig. 3.5.

ALGORITHM 3.1.4 (2D INVERSE COMPLEX WAVELET TRANSFORM)

- *At level $j > 1$, each tree is processed independently. The symbols $ee...oe$ represent the inverse of the decimation, for example $\overset{ee}{x}_{n,m} = x_{p,q}$ if $n = 2p$ and $m = 2q$, 0 in the other cases.*

$$
\begin{aligned}
a_A^j &= (\overset{ee}{a})_A^{j+1} \star \tilde{h}^e\tilde{h}^e + (\overset{ee}{d})_A^{j+1,1} \star \tilde{g}^e\tilde{h}^e + (\overset{ee}{d})_A^{j+1,2} \star \tilde{h}^e\tilde{g}^e + (\overset{ee}{d})_A^{j+1,3} \star \tilde{g}^e\tilde{g}^e \\
a_B^j &= (\overset{eo}{a})_B^{j+1} \star \tilde{h}^e\tilde{h}^o + (\overset{eo}{d})_B^{j+1,1} \star \tilde{g}^e\tilde{h}^o + (\overset{eo}{d})_B^{j+1,2} \star \tilde{h}^e\tilde{g}^o + (\overset{eo}{d})_B^{j+1,3} \star \tilde{g}^e\tilde{g}^o \\
a_C^j &= (\overset{oe}{a})_C^{j+1} \star \tilde{h}^o\tilde{h}^e + (\overset{oe}{d})_C^{j+1,1} \star \tilde{g}^o\tilde{h}^e + (\overset{oe}{d})_C^{j+1,2} \star \tilde{h}^o\tilde{g}^e + (\overset{oe}{d})_C^{j+1,3} \star \tilde{g}^o\tilde{g}^e \\
a_D^j &= (\overset{oo}{a})_D^{j+1} \star \tilde{h}^o\tilde{h}^o + (\overset{oo}{d})_D^{j+1,1} \star \tilde{g}^o\tilde{h}^o + (\overset{oo}{d})_D^{j+1,2} \star \tilde{h}^o\tilde{g}^o + (\overset{oo}{d})_D^{j+1,3} \star \tilde{g}^o\tilde{g}^o
\end{aligned}
$$

- *At level 1, the 4 trees are averaged :*

$$
\left.
\begin{aligned}
a^0 = \quad & \big[(\overset{ee}{a})_A^1 + (\overset{eo}{a})_B^1 + (\overset{oe}{a})_C^1 + (\overset{oo}{a})_D^1\big] \star \tilde{h}^o\tilde{h}^o \\
& + \big[(\overset{ee}{d})_A^{1,1} + (\overset{eo}{d})_B^{1,1} + (\overset{oe}{d})_C^{1,1} + (\overset{oo}{d})_D^{1,1}\big] \star \tilde{g}^o\tilde{h}^o \\
& + \big[(\overset{ee}{d})_A^{1,2} + (\overset{eo}{d})_B^{1,2} + (\overset{oe}{d})_C^{1,2} + (\overset{oo}{d})_D^{1,2}\big] \star \tilde{h}^o\tilde{g}^o \\
& + \big[(\overset{ee}{d})_A^{1,3} + (\overset{eo}{d})_B^{1,3} + (\overset{oe}{d})_C^{1,3} + (\overset{oo}{d})_D^{1,3}\big] \star \tilde{g}^o\tilde{h}^o
\end{aligned}
\right\} \times \frac{1}{4}
$$

3.1.3 Advantages of complex wavelets

Limited redundancy, computational efficiency

In the unidimensional case, the redundancy is 2:1 at the first scale since a second tree is introduced, and in the bidimensional case, the redundancy becomes 4:1. As the 4 trees are nearly classical wavelet trees beyond level 1, this value remains the same for the whole transform, independent of the scale.

Thus, the computational complexity is only increased by 3 compared to classical wavelet transforms. The algorithm remains fast, since the discrete transform can be implemented by a ladder filter structure.

Shift invariance

First, the dual-tree transform is designed to be, as much as possible, translation invariant if we study the reconstructed image. It means that if we decide to keep only the details or the approximation of a given scale, removing all other scales, shifting the input image only produces a shift of the reconstructed filtered image, without aliasing.

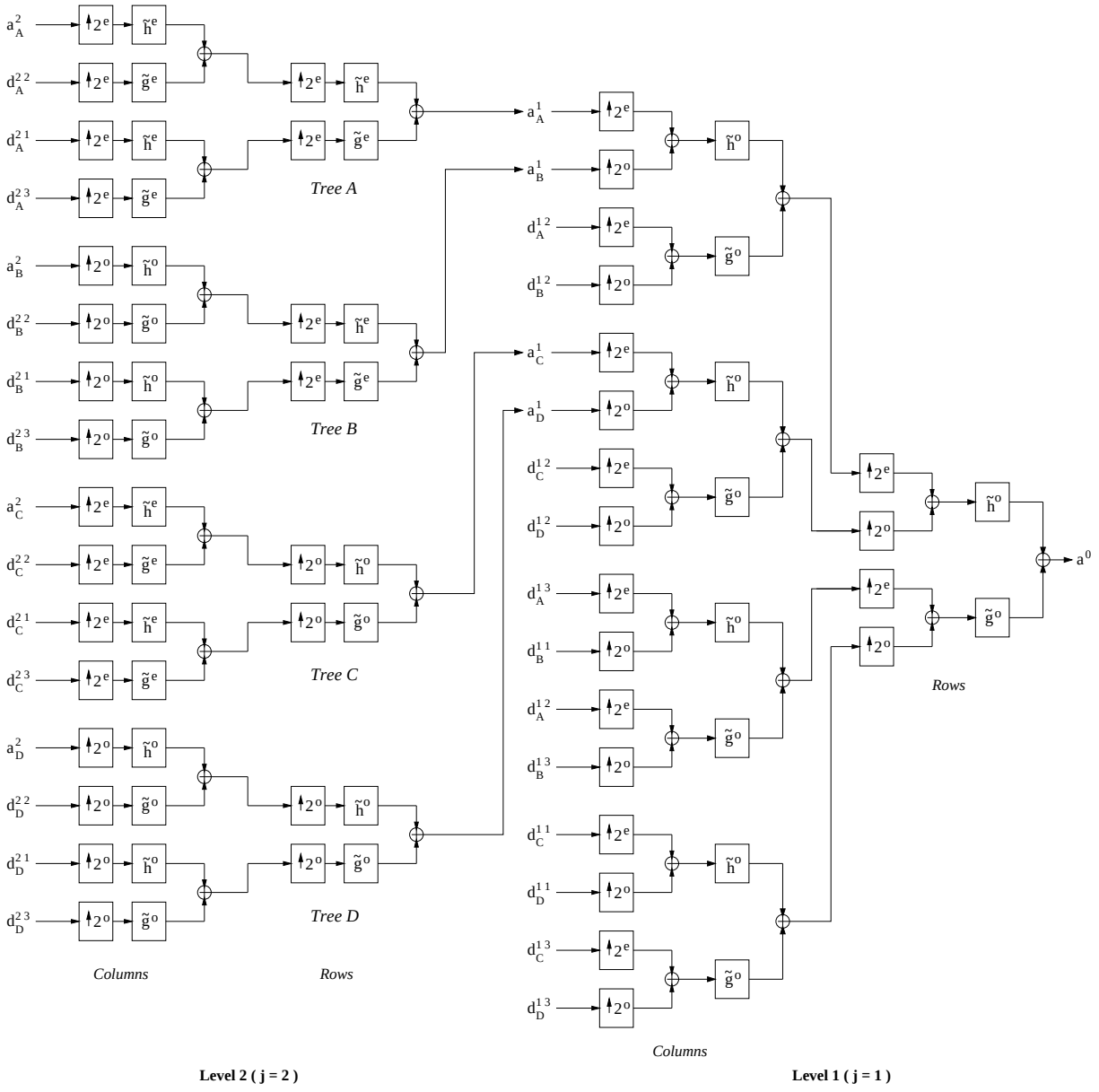


Figure 3.5: *Quad-tree filter bank for the bidimensional inverse complex wavelet transform*

Second, this transform is designed to make the total energy of a given subband approximately invariant to translations of the input image. The consequence is the subbands are shifted by a fraction of pixel when the input image is translated by an integer number of pixels. Of course, these coefficients are not translation invariant but their amplitude does not oscillate through zero, unlike real wavelets. Therefore, if they are thresholded, the translation produces less artefacts than in the real case.

Good directional selectivity

One of the most important properties of the complex transforms is the ability to separate more directions than real separable wavelet transforms. For each pixel of the detail images, the 4 coefficients corresponding to the 4 parallel trees are converted into 2 complex numbers. At each scale, since there are 3 detail images, it gives 6 complex subbands, as illustrated by Fig. 3.6. If we examine the Fourier transform of the complex impulse response corresponding to these subbands, we find that they partition the space into 6 distinct directions, as illustrated in Fig. 3.7. This figure also shows the subband labeling.

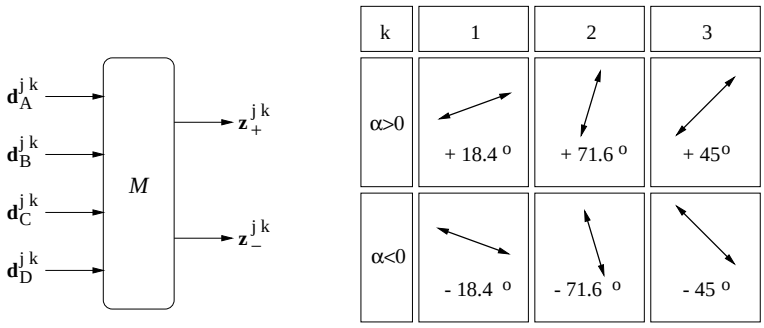


Figure 3.6: *Left : conversion between quad-tree real details $d^{j\,k}$ and complex oriented details $z_{\pm}^{j}$ using (3.5), right : orientation α corresponding to different subbands k, given by equ. (3.6)*

This directional separation is made possible by using filters whose frequency responses suppress most of the negative frequencies, even if the bidimensional transform is separable.

To correctly represent a real bidimensional signal, one half of its spectrum is required, since the spectrum exhibits a symmetry w.r.t. the (0,0) frequency. It means that two adjacents quadrants are required.

The problem with real separable wavelets is that the basis vectors have a spectral response which is symmetric w.r.t. both axes, which avoids separation of positive and negative frequencies. That is why real wavelets do not represent correctly the diagonal features : the diagonal subbands capture both negative and positive frequencies, therefore they mix both $+45^o$ and -45^o orientations.

In the complex transform, the filters have an asymmetric response, they can efficiently reject negative frequencies. Therefore, the complex conjugate filters are also used in the

filter bank, because they have a symmetric response w.r.t. the original filters (if one filter selects positive frequencies, its conjugate selects negative frequencies).

Thus, negative and positive frequencies can be separated vertically and horizontally, which provides a strong orientation of the impulse responses (see Fig. 3.8), and therefore a high directional selectivity between the different subbands. Thus, positive and negative diagonal orientations are not mixed anymore. Fig. 3.9 shows the transform of an isotropic synthetic image, which also contains details at different scales. The orientation selectivity is clear, and completes the classical scaling property of this wavelet transform.

Thus, the variously oriented features present in an image can be efficiently represented. It provides a sort of rotation invariance, since the energy of an oriented feature at a given scale is not very sensitive to a rotation of the input image, that completes the approximate translation invariance presented before.

Furthermore, an efficient representation of a textured image enables better separation of the signal from the noise, if we focus on the thresholding of noisy coefficients. As the energy is conserved between an image and its transform, a better concentration of the information associated with an oriented texture means higher signal coefficients in the corresponding subband. At the same time, most noise has an isotropic distribution, and remains equally distributed over all subbands of a given scale.

3.2 Necessity of using complex wavelet packets

The complex wavelet basis has been successfully used for image denoising [33]. The white noise is uniformly distributed across the subbands, while the signal transform (see Fig. 3.10 for an example) concentrates over a small number of coefficients.

If we consider the deconvolution as a denoising problem, we have to deal with *colored* noise, and the complex wavelet basis no longer allows the covariance matrix K of the deconvolved noise to be diagonal. It is shown on Fig. 3.11 that the high frequency subbands are contaminated by the deconvolved noise, so that the signal present in these subbands is not recoverable by any thresholding method.

We construct a *complex wavelet packet* basis, by analogy with real wavelet packets [29, 31]. However, it has not been proven yet that a complex basis gives approximate eigenvectors of the operator K. See Fig. 3.12 for an experimental verification of this property : the correlation is strongly reduced between wavelets and wavelet packets.

To limit the growth of the deconvolved noise variance at high frequencies, the finest scale subbands are decomposed, in the same manner as the lowest scales in a classical wavelet transform. It avoids the “explosion” of the noise, since the wavelet packets have a frequential support which decreases exponentially at high frequencies.

We expect from the complex basis to exhibit the same property. We can examine each tree separately, and we remark that it should remain true for each one. For tree D, which corresponds to a real biorthogonal wavelet transform, it has been shown that the corresponding biorthogonal wavelet basis achieves an approximate diagonalization of K [29, 32].

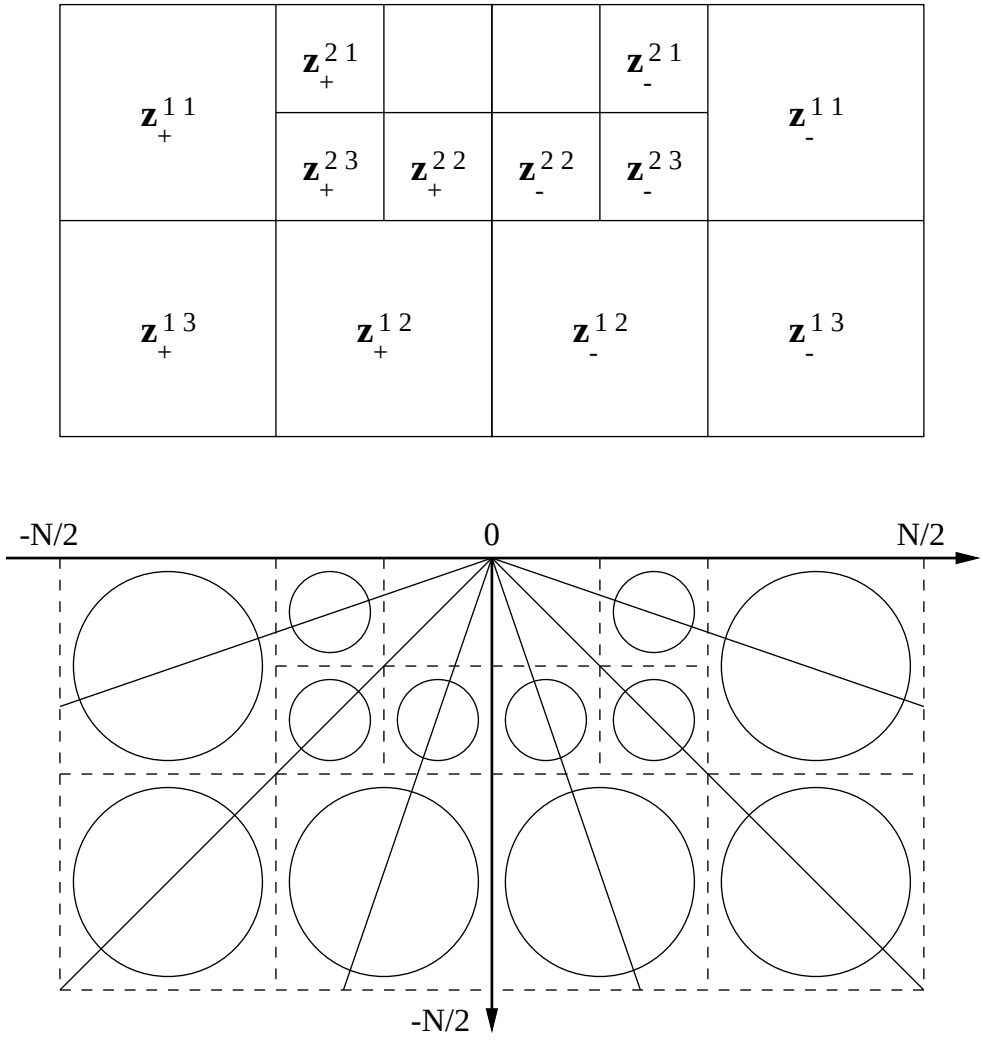
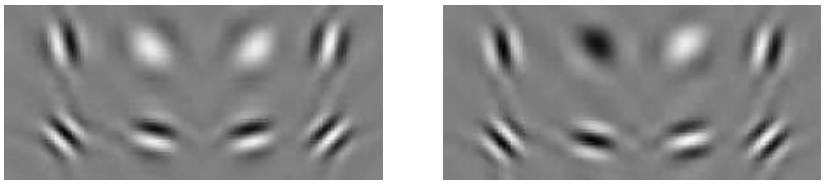


Figure 3.7: *Top : labeling of the different subbands, bottom : rough partition of the frequency space corresponding to the complex wavelet transform, showing the directional selectivity*



Real part Imaginary part

Figure 3.8: *Impulse response of the complex wavelet transform at level 4, showing the ability to separate 6 directions*

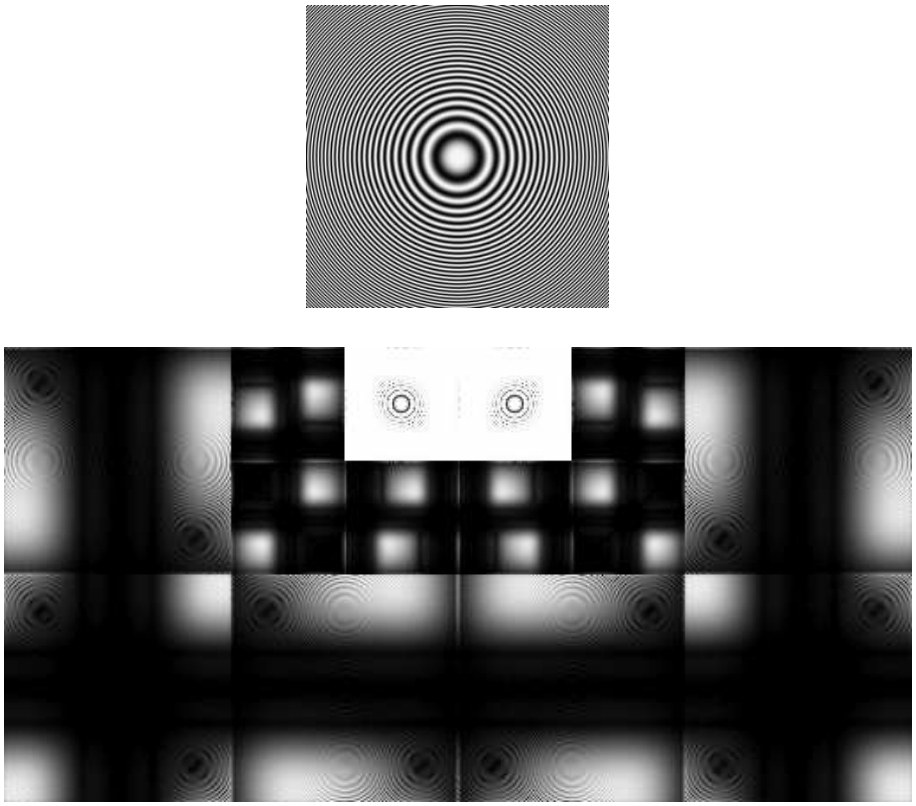


Figure 3.9: *Top : isotropic test image containing various scale information, bottom : magnitude of its complex wavelet transform showing both directional and scaling properties*

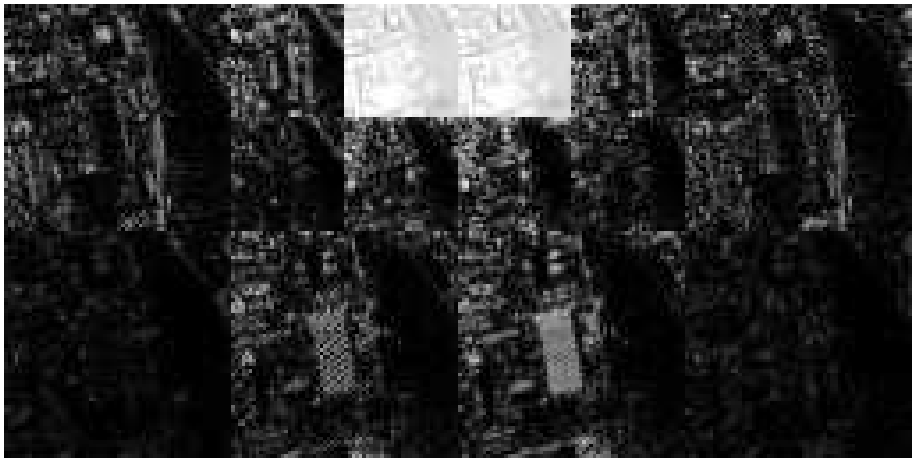


Figure 3.10: *Magnitude of the complex wavelet transform of the original image of Fig. 1.1*

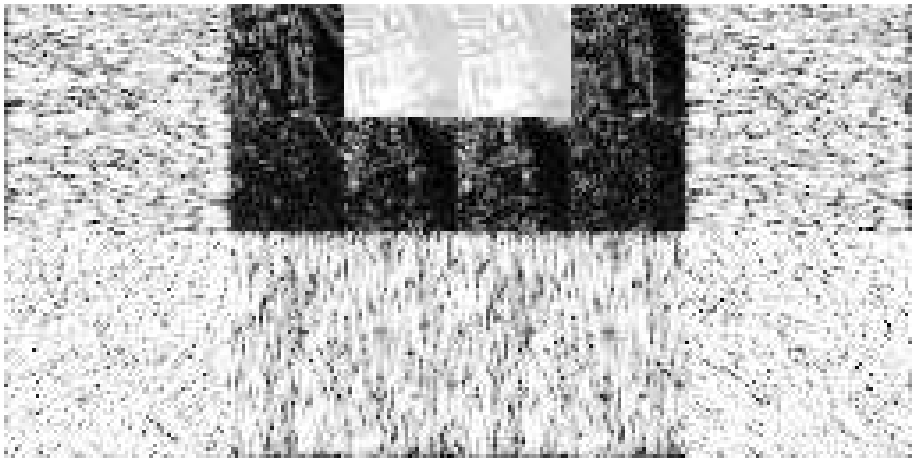


Figure 3.11: *Magnitude of the complex wavelet transform of the image of Fig. 1.1 deconvolved without regularization : the noise "explodes" in high frequency subbands*

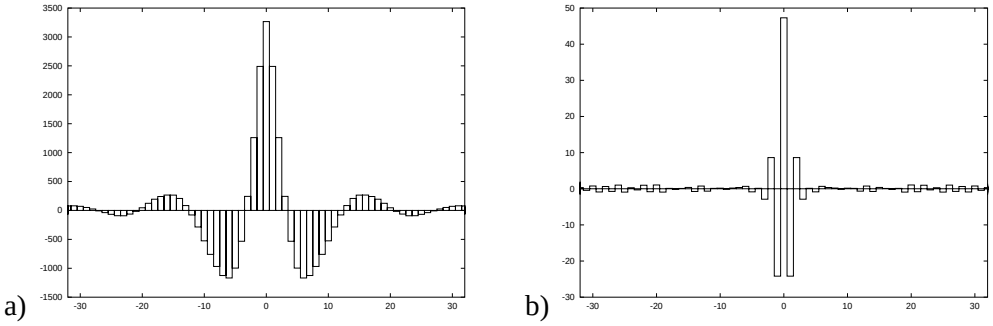


Figure 3.12: *Real part of the covariance of the deconvolved noise subbands (w.r.t. columns) a) Complex wavelets : (1,1,0) b) Complex wavelet packets : (2,2,0)*

3.2.1 Bidimensional complex wavelet packet transform

The complex wavelet packet filter bank

See the previous chapter, section 2.2, for the definition of the wavelet packet transform and the properties of the associated quad-tree.

Instead of splitting only the approximations a at a given scale into details d and a larger scale approximation, it is possible to also decompose the details. It corresponds to splitting a detail space W_j into new spaces, deriving new bases.

This operation is done beyond level 1, so it is achieved in each tree independently. Therefore, for each tree A to D, a quad-tree is constructed by applying a separable filter bank, the same that is used to compute the complex wavelet coefficients for $j > 1$. Thus, the bidimensional frequency space is divided into square regions of similar sizes, as illustrated by Fig. 3.16.

Let (p, q) denote the node of a quad-tree at a given depth j. We can construct a discrete wavelet packet basis of $\mathrm{l}^2(\mathbb{R}^2)$ related to each tree. We denote $d_T^{j,p,q}$ the subband related to the node (p, q) at level j, in the quad-tree T, where $T \in \{A, B, C, D\}$. Depending on p and q, it can be an approximation, a wavelet detail, or a wavelet packet detail.

The following algorithm (illustrated by Fig. 3.14) computes the complex wavelet packet transform. The discrete bidimensional basis associated to this decomposition is detailed in appendix B.

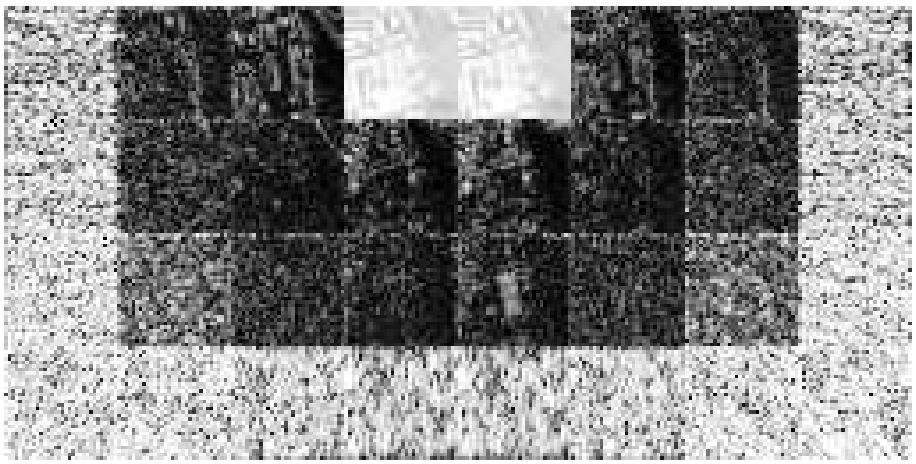


Figure 3.13: *Magnitude of the complex wavelet packet transform of the image of Fig. 1.1 deconvolved without regularization, showing the ability to recover high frequency information (to be compared with Fig. 3.11)*

ALGORITHM 3.2.1 (2D COMPLEX WAVELET PACKET TRANSFORM)
Use the 2D complex wavelet transform (Algo. 3.1.3) for the first level, and the following decomposition for $j > 1$:

Tree T	**A**	**B**	**C**	**D**
$(d_T^{j+1,2p,2q})_n$	$(d_A^{j,p,q} \star h^e h^e)_{e,e}$	$(d_B^{j,p,q} \star h^e h^o)_{e,o}$	$(d_C^{j,p,q} \star h^o h^e)_{o,e}$	$(d_D^{j,p,q} \star h^o h^o)_{o,o}$
$(d_T^{j+1,2p+1,2q})_n$	$(d_A^{j,p,q} \star g^e h^e)_{e,e}$	$(d_B^{j,p,q} \star g^e h^o)_{e,o}$	$(d_C^{j,p,q} \star g^o h^e)_{o,e}$	$(d_D^{j,p,q} \star g^o h^o)_{o,o}$
$(d_T^{j+1,2p,2q+1})_n$	$(d_A^{j,p,q} \star h^e g^e)_{e,e}$	$(d_B^{j,p,q} \star h^e g^o)_{e,o}$	$(d_C^{j,p,q} \star h^o g^e)_{o,e}$	$(d_D^{j,p,q} \star h^o g^o)_{o,o}$
$(d_T^{j+1,2p+1,2q+1})_n$	$(d_A^{j,p,q} \star g^e g^e)_{e,e}$	$(d_B^{j,p,q} \star g^e g^o)_{e,o}$	$(d_C^{j,p,q} \star g^o g^e)_{o,e}$	$(d_D^{j,p,q} \star g^o g^o)_{o,o}$

As seen in Fig. 3.13, a part of the highest frequency subbands becomes recoverable, whereas these subbands are fully contaminated by noise in the case of the simple wavelet transform (see Fig. 3.11). The noise is now concentrated on the borders of the frequency space and allows the SNR to be higher in other regions. In spite of its apparent complexity, the algorithm remains faster than the *shift invariant* real decomposition. The reason is that to enable such properties, real transforms have to be averaged over a minimum of 8 integer translations, because they generally are translation variant. See Sections 3.4.3 and 3.5.2 for a comparison of computational complexity of complex and real transforms.

3.2.2 Improved directional selectivity

The complex wavelet packets enable one to divide the frequency space into more directions than the wavelets. Indeed, instead of dividing an approximation space which defines no particular orientation, the wavelet packet decomposition processes the detail subbands, which are strongly oriented. Each detail subband at level 1 isolates an area of the frequency space defined by a mean direction and a dispersion, enabling one to select a range of directions around an orientation α. If the subband is decomposed into 4 new subbands, it means that the corresponding frequency area splits into 4 new areas, which can define new orientations, as shown in Fig. 3.16. Thus, the impulse responses corresponding to the different subbands, illustrated by Fig. 3.17, are strongly oriented.

The transform of the synthetic image of Fig. 3.9, given in Fig. 3.18, illustrates how the detail spaces of the level 1 provide new directions at level 2. If these directions are averaged, we find again the directions of the previous level.

Thus, the directional selectivity is improved by this new decomposition. In fact, the number of different directions created this way depends on the level of the transform. To compute the angles associated with the subbands and to enumerate the directions, let us divide the frequency space into squares, w.r.t. the quad-tree structure of the algorithm. If we consider that the angle is given by the center of the squares, we find the following relation :

$$\alpha(p,q) = \arctan\left(\frac{2q+1}{2p+1}\right) \tag{3.6}$$

which is used to construct the table of Fig. 3.15, showing up to 26 different directions for the level 2 transform.

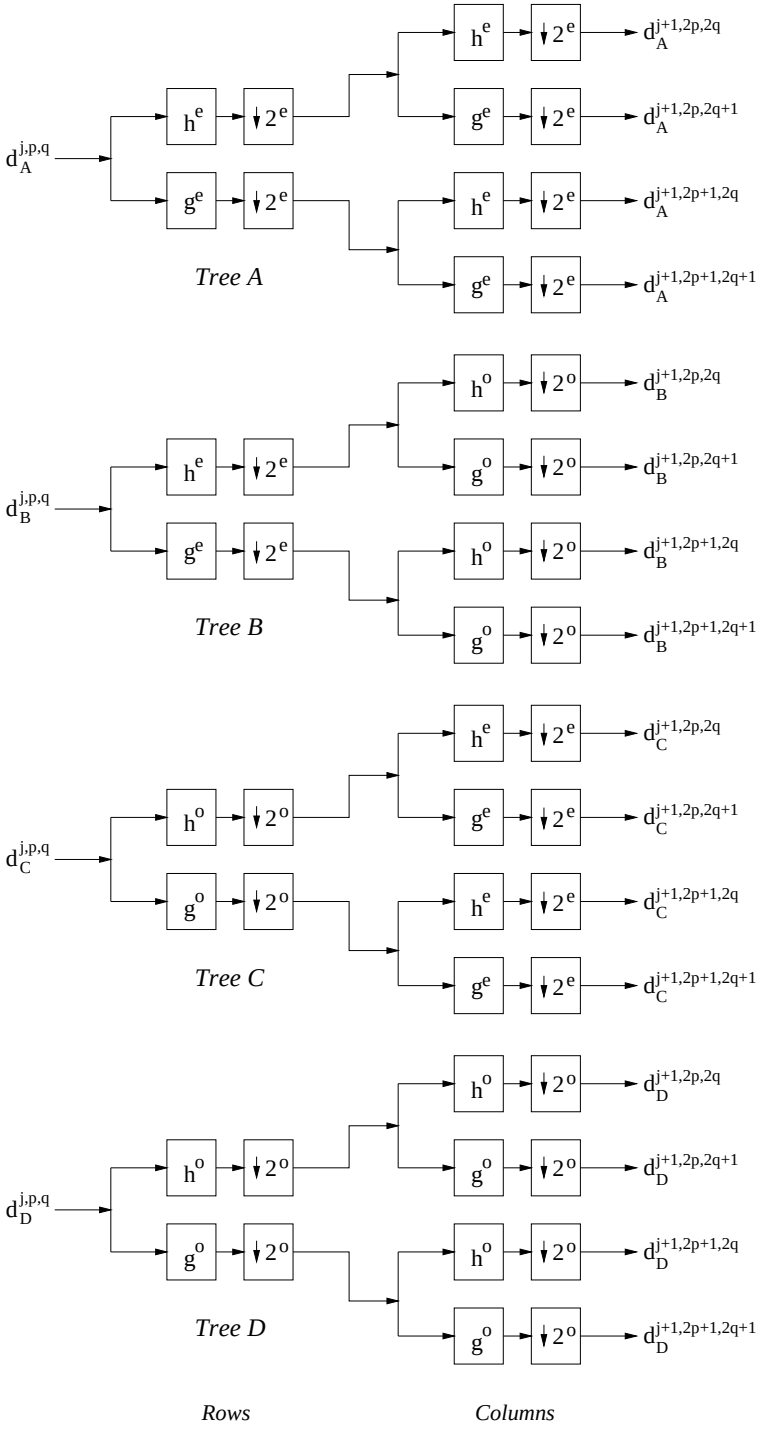


Figure 3.14: *Quad-tree filter bank for the bidimensional complex wavelet packet transform (for $j > 1$)*

q \ p	0	1	2	3
0	-	18.43^o	11.31^o	08.13^o
1	71.57^o	45.00^o	30.96^o	23.20^o
2	78.69^o	59.03^o	45.00^o	35.54^o
3	81.87^o	66.80^o	54.46^o	45.00^o

Figure 3.15: *Orientation angle corresponding to the complex wavelet and complex wavelet packet subbands (p,q), for $j = 2$ - wavelet subbands 1,2,3 correspond to packet coefficients $(1,0),(0,1),(1,1)$*

Beyond level 2, this number depends on the tree $\mathbf{Q}$. In the following, we will choose a tree which only divides the highest and lowest frequency subbands, as in the real case, since it gives optimal results for deconvolution.

The meaning of such a number is that oriented features are classified into different subbands, depending on their orientation. Thus, a signal concentrates over fewer coefficients than with other transforms (such as real wavelets), and becomes easier to separate from noise. Consider the striped roof of the original image of Fig. 1.1 and its transform in Fig. 3.19. In spite of the deconvolved noise added in Fig. 3.13, this feature can be extracted thanks to its strong orientation.

3.2.3 Bidimensional inverse complex wavelet packet transform

The transform described by Algorithm 3.2.1, beside its useful directional and invariance properties, is perfectly invertable. The reconstruction is given by the following filter bank, illustrated by Fig. 3.20.

ALGORITHM 3.2.2 (2D INVERSE COMPLEX WAVELET PACKET TRANSFORM)
Each tree is processed independently. The symbols $ee...oe$ represent the inverse of the decimation, for example $\overset{ee}{x}_{n,m} = x_{p,q}$ if $n = 2p$ and $m = 2q$, 0 in the other cases.

$$
\begin{aligned}
d_A^{j,p,q} =& (\overset{ee}{d})_A^{j+1,2p,2q} \star\tilde{h}^e\tilde{h}^e + (\overset{ee}{d})_A^{j+1,2p+1,2q} \star\tilde{g}^e\tilde{h}^e \\
&+ (\overset{ee}{d})_A^{j+1,2p,2q+1} \star\tilde{h}^e\tilde{g}^e + (\overset{ee}{d})_A^{j+1,2p+1,2q+1} \star\tilde{g}^e\tilde{g}^e \\
a_B^{j,p,q} =& (\overset{eo}{d})_B^{j+1,2p,2q} \star\tilde{h}^e\tilde{h}^o + (\overset{eo}{d})_B^{j+1,2p+1,2q} \star\tilde{g}^e\tilde{h}^o \\
&+ (\overset{eo}{d})_B^{j+1,2p,2q+1} \star\tilde{h}^e\tilde{g}^o + (\overset{eo}{d})_B^{j+1,2p+1,2q+1} \star\tilde{g}^e\tilde{g}^o \\
a_C^{j,p,q} =& (\overset{oe}{d})_C^{j+1,2p,2q} \star\tilde{h}^o\tilde{h}^e + (\overset{oe}{d})_C^{j+1,2p+1,2q} \star\tilde{g}^o\tilde{h}^e \\
&+ (\overset{oe}{d})_C^{j+1,2p,2q+1} \star\tilde{h}^o\tilde{g}^e + (\overset{oe}{d})_C^{j+1,2p+1,2q+1} \star\tilde{g}^o\tilde{g}^e \\
a_D^{j,p,q} =& (\overset{oo}{d})_D^{j+1,2p,2q} \star\tilde{h}^o\tilde{h}^o + (\overset{oo}{d})_D^{j+1,2p+1,2q} \star\tilde{g}^o\tilde{h}^o \\
&+ (\overset{oo}{d})_D^{j+1,2p,2q+1} \star\tilde{h}^o\tilde{g}^o + (\overset{oo}{d})_D^{j+1,2p+1,2q+1} \star\tilde{g}^o\tilde{g}^o
\end{aligned}
$$

$\mathbf{z}_{+}^{2\,30}$	$\mathbf{z}_{+}^{2\,20}$	$\mathbf{z}_{+}^{2\,1}$			$\mathbf{z}_{-}^{2\,1}$	$\mathbf{z}_{-}^{2\,20}$	$\mathbf{z}_{-}^{2\,30}$
$\mathbf{z}_{+}^{2\,31}$	$\mathbf{z}_{+}^{2\,21}$	$\mathbf{z}_{+}^{2\,3}$	$\mathbf{z}_{+}^{2\,2}$	$\mathbf{z}_{-}^{2\,2}$	$\mathbf{z}_{-}^{2\,3}$	$\mathbf{z}_{-}^{2\,21}$	$\mathbf{z}_{-}^{2\,31}$
$\mathbf{z}_{+}^{2\,32}$	$\mathbf{z}_{+}^{2\,22}$	$\mathbf{z}_{+}^{2\,12}$	$\mathbf{z}_{+}^{2\,02}$	$\mathbf{z}_{-}^{2\,02}$	$\mathbf{z}_{-}^{2\,12}$	$\mathbf{z}_{-}^{2\,22}$	$\mathbf{z}_{-}^{2\,32}$
$\mathbf{z}_{+}^{2\,33}$	$\mathbf{z}_{+}^{2\,23}$	$\mathbf{z}_{+}^{2\,13}$	$\mathbf{z}_{+}^{2\,03}$	$\mathbf{z}_{-}^{2\,03}$	$\mathbf{z}_{-}^{2\,13}$	$\mathbf{z}_{-}^{2\,23}$	$\mathbf{z}_{-}^{2\,33}$

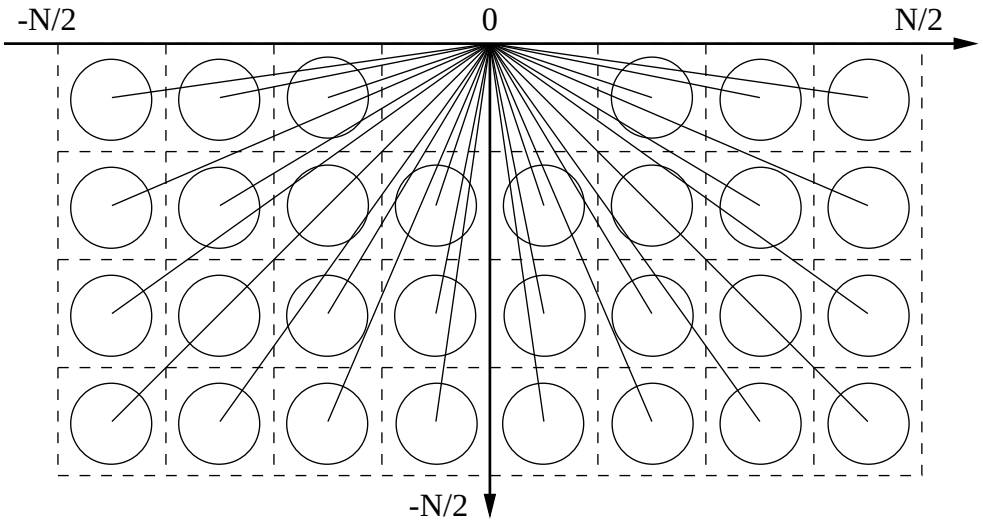


Figure 3.16: *Top : labeling of the different subbands, bottom : rough partition of the frequency space corresponding to the complex wavelet packet transform, showing the directional selectivity*

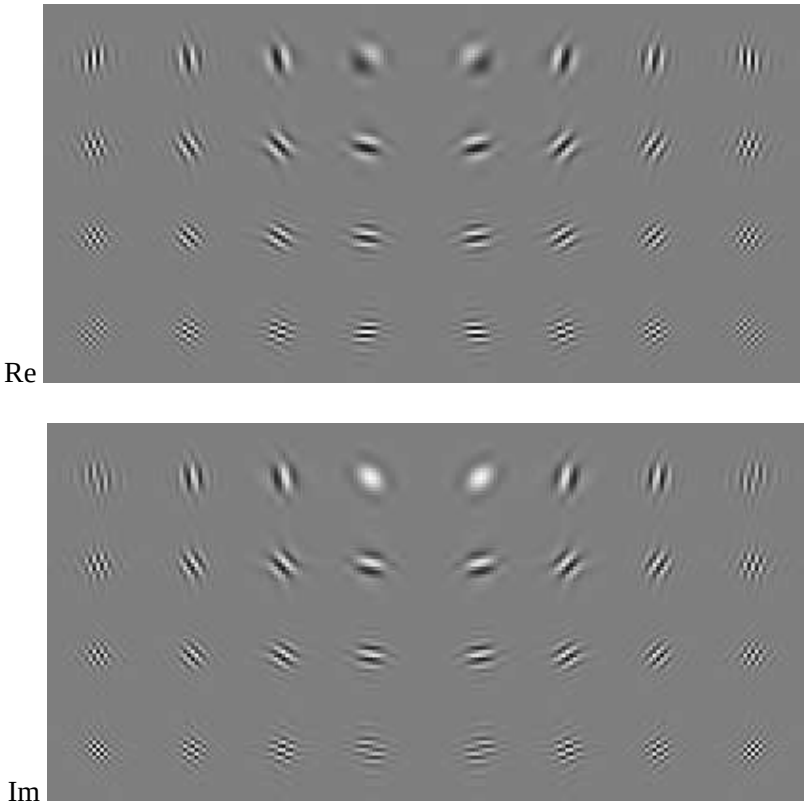


Figure 3.17: *Impulse response of the complex wavelet packet transform at level 2, showing the ability to separate 26 directions*

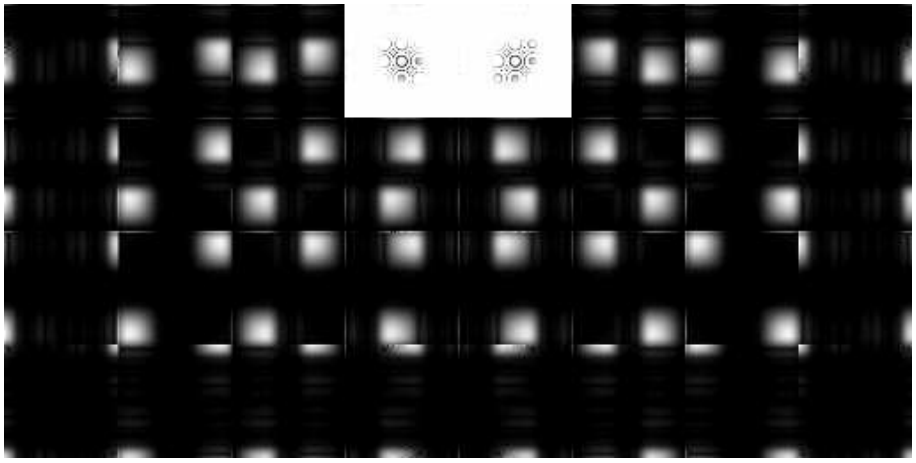


Figure 3.18: *Magnitude of the complex wavelet packet transform of the test image of Fig. 3.9, showing both directional and scaling properties*

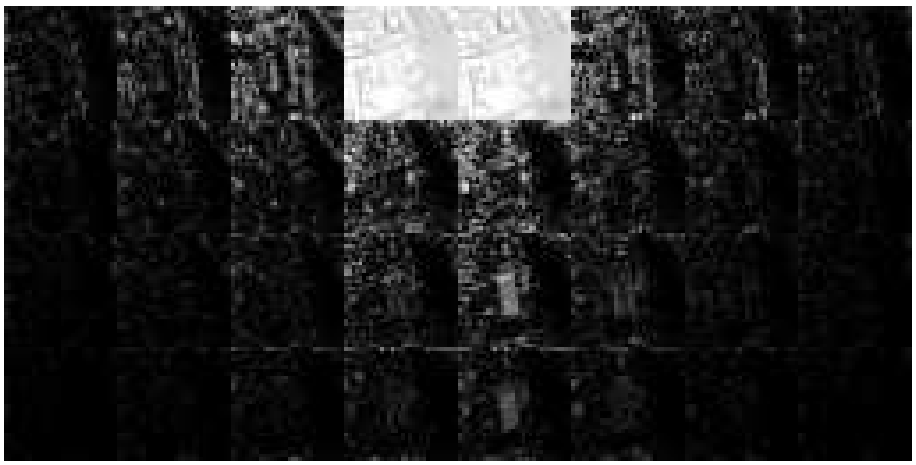


Figure 3.19: *Magnitude of the complex wavelet packet transform of the original image of Fig. 1.1*

We have defined a framework enabling to build a thresholding algorithm, by computing a forward and a reverse transform, associated to a basis which provides a better signal representation than the real wavelet packet basis.

3.3 Optimal threshold estimation

3.3.1 Minimal risk of thresholding estimators

We have detailed a basis choice which is near optimal for noise filtering (see discussion of Section 1.2). We have now to determine which is the best choice for the threshold value associated with the attenuation function used to estimate the unknown transform coefficients.

We refer to section 2.3.1 for the definitions of x, ξ, and n, respectively denoting the noisy, original, and noise transform coefficients. We denote a_k the attenuation function for the subband k. These variables are complex, since the basis $\mathcal{B}$ is related to the complex wavelet packets.

To obtain the expression of the noise coefficients n, let us recall equ. (3.5). Each complex coefficient is obtained by summing or subtracting the coefficients of the 4 trees A, B, C, D. If we compute the covariance between the real and imaginary parts, we find (for the coefficients z_+) $E[n_r n_i] = E[n_A n_B + n_A n_C - n_D n_B - n_D n_C]$ where n_A, n_B, n_C, n_D, are the noise coefficients for each tree. By symmetry assumptions, this covariance is null, since all covariances $E[n_T n_{T'}]$ between different trees T and T' are equal. Then, the distribution of the noise is defined as a joint distribution of (n_r, n_i) :

$$P(n) = \frac{1}{2\pi\,\sigma_k^2} e^{-|n|^2/2\sigma_k^2} \tag{3.7}$$

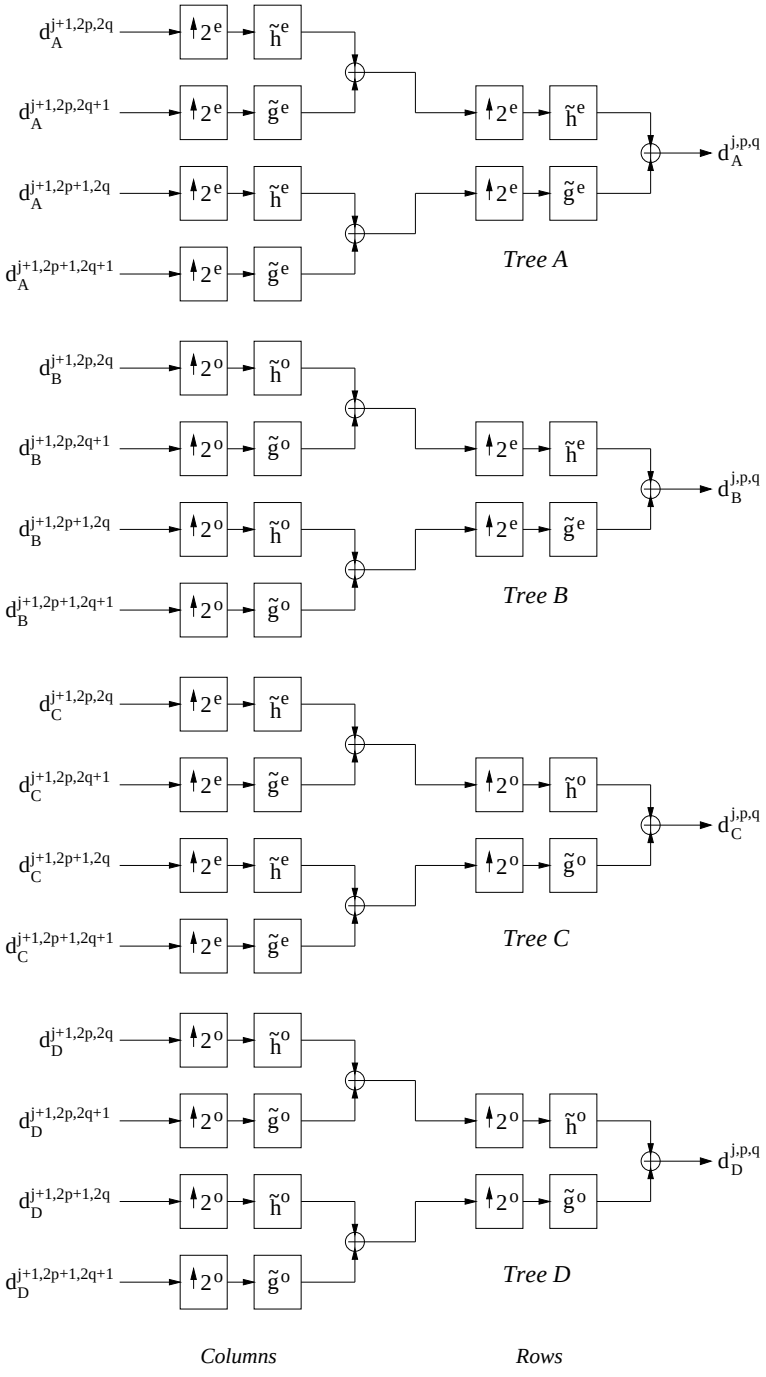


Figure 3.20: *Quad-tree filter bank for the bidimensional inverse complex wavelet packet transform (for $j > 1$)*

We will again present two methods to compute the threshold T ; the first is based on a minimax risk calculus [32], and the second tries to optimize the SNR by modeling the image transform complex subbands.

Refer to equ. (2.3) for a definition of the risk of an estimator $\hat{X}$. Since we deal with complex signals, it is rewritten as :

$$r(\hat{X}, \mathcal{X}) = E\left[\|\hat{X} - \mathcal{X}\|^2\right] = E\left[\sum_{k,m} \left| a_k(x) - \xi \right|^2\right] \tag{3.8}$$

The minimum risk is achieved by the following attenuation, called oracle attenuation because in practice the original image $\mathcal{X}$ is not known :

$$a_k(\langle X, g_m \rangle) = \frac{|\xi|^2}{2\sigma_m^2 + |\xi|^2}\, x \tag{3.9}$$

(To derive this equation, simply take the derivative of the risk (3.8) w.r.t. a_k and set it to zero). The factor 2 which multiplies σ_m^2 is due to the complex nature of the coefficients (dimension 2).

This attenuation consists of filtering only the *magnitude* of the complex coefficients. It enables us to preserve the properties of the complex transform, since the magnitude is not very sensitive to any translation of the image. Such an operation preserves the shift invariance property of the transform, which avoids artefacts which appear with non shift-invariant real transforms. The magnitudes are obtained by equ. (3.5). They are thresholded, and then equ. (3.5) is inverted to return to the 4 parallel trees.

An upper bound of the risk has been computed in [17], in the case of real signals, by choosing a threshold operator $a = \theta_T$ (see equ. (2.21)), with $T = \sigma_m \sqrt{2 \log_e N_x N_y}$. If we deal with complex signals and a complex thresholding function, we have no knowledge about upper bound computation in this case.

We redefine the thresholding functions in the complex case, by applying the functions (2.21) to the magnitude of the complex variables x :

$$\theta_T^s(x) = \begin{cases} x(1 - T/|x|) & \text{if } |x| > T \\ 0 & \text{if } |x| \leq T \end{cases} \qquad \theta_T^h(x) = \begin{cases} x & \text{if } |x| > T \\ 0 & \text{if } |x| \leq T \end{cases} \tag{3.10}$$

A thresholding method can be constructed by using equ. (3.9) and trying to predict the unknown coefficients ξ. If we consider that a good prediction is achieved by $\xi \simeq a(x)$, then we have to solve :

$$a(x) = \frac{|a(x)|^2}{2\sigma_k^2 + |a(x)|^2} x \tag{3.11}$$

which gives multiple solutions. We retain the following function, since it is equal to identity when there is no noise ($\sigma_k = 0$). We call it "oracle threshold" :

$$\theta_T^o(x) = \begin{cases} \frac{x}{2}\left(1 + \sqrt{1 - |T/x|^2}\right) & \text{if } |x| > T \\ 0 & \text{if } |x| \leq T \end{cases} \tag{3.12}$$

$$\text{with} \quad T = \begin{cases} 2\sqrt{2}\sigma_k & \text{if } \sigma_k < \sup_{\mathcal{X}\in\Omega}|\xi| \\ \infty & \text{if } \sigma_k \geq \sup_{\mathcal{X}\in\Omega}|\xi| \end{cases} \tag{3.13}$$

This expression naturally adapts to the variance of the noise in each subband. However, this approach does not take into account the statistics of the unknown image. These methods correspond to the least favourable prior case, i.e. they should work with any type of input image. Here again, we make the same remark as in the real case : the drawback is removing too much of the signal.

We can ask if these methods are really optimal, or if we could find another one which would provide a smaller risk. By taking into account the prior distribution of the data, a more accurate thresholding should be possible. Therefore, a model of the image subbands is needed.

3.3.2 Modeling the original image subbands

Real wavelet coefficients can be represented by Generalized Gaussian distributions [1, 2, 37, 43]. It is possible to model each complex wavelet packet subband by a homogeneous GGMRF, as in the real case, to construct a threshold estimation method. It has been verified experimentally (see Fig. 3.21) that the histogram of an original image can be modeled by :

$$P(\xi) = \frac{1}{Z(\alpha_k, p_k)}\, e^{-|\xi/\alpha_k|^{p_k}} \tag{3.14}$$

where $|\xi/\alpha_k|$ represents a magnitude of a complex number. As shown by Fig. 3.22, the complex density is a bidimensional function which only depends on the magnitude (it exhibits a radial symmetry). It is generally not separable for $p \neq 2$. We set a given density function on the magnitude, while the phase is uniformly distributed in $[0, 2\pi]$.

We assume that the coefficients ξ are independent in a given subband, between different subbands of a given scale, and also between scales. This is an approximation which enables a fast thresholding technique, we will not handle here possible correlations between subbands or neighbour coefficients.

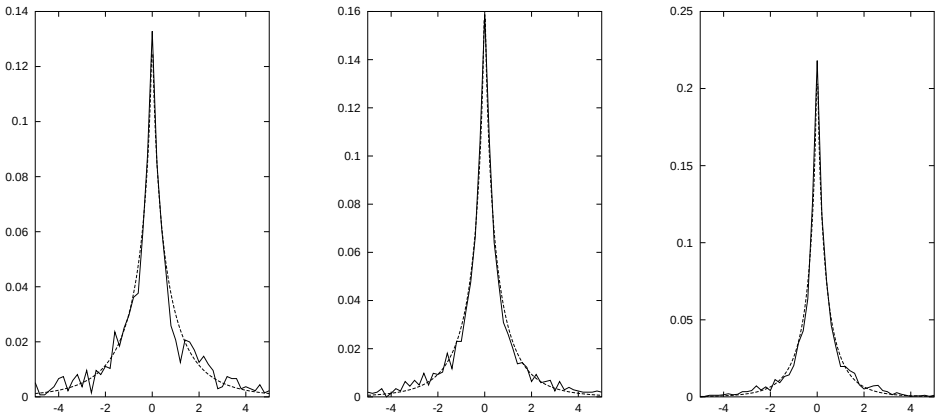
The exponent p_k can be set to a fixed value, the same for all subbands, to simplify the computation. Indeed, if this parameter is not specified, it must be estimated which is quite complex and is probably not justified by the improvement of the results.

On large size images, the prior parameter of a GGMRF model can be efficiently computed by various methods, such as ML for example. We choose to estimate the parameter α of the distribution from the histogram of a given subband, as in the real case.

The covariance matrix of the noise is supposed to be nearly diagonal in the chosen basis (see Fig. 3.12). So we consider that the noise variables are independent in the wavelet packet basis.

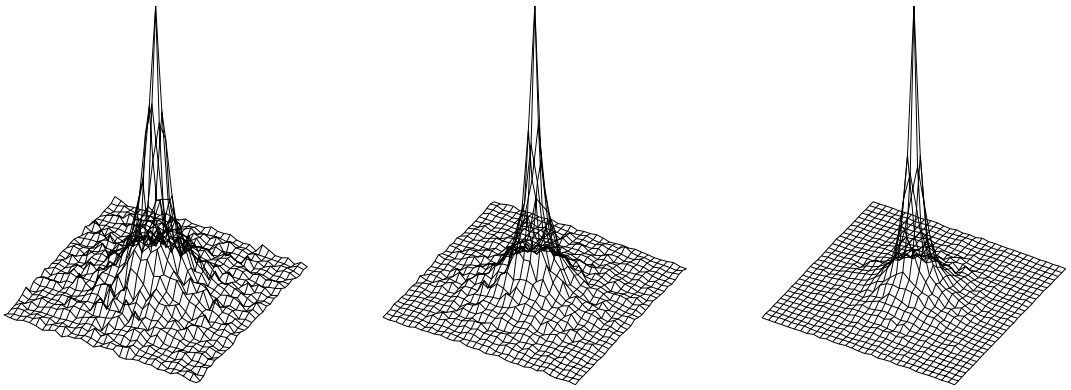
The threshold selection can be made to minimize the risk for each subband k :

$$\hat{T}(\alpha_k, \sigma_k) = \arg\inf_{T}\ E\left[|\theta_T(\xi + n) - \xi|^2\right] \tag{3.15}$$



(2,2,0)+ (2,2,1)+ (2,2,2)+

Figure 3.21: *Conditional distributions $P(\xi \mid \xi \in \mathbb{R})$ of complex wavelet packet coefficients of the original image of Fig. 4.6 for subbands (j, p, q) ; the dashed curve corresponds to the prior model (3.14) with $p = 0.7$*



(2,2,0)+ (2,2,1)+ (2,2,2)+

Figure 3.22: *Distributions $P(\xi)$ of complex wavelet packet coefficients of the original image of Fig. 4.6 for subbands (j, p, q)*

where θ_T is one of the three thresholding functions defined by (2.21) and (2.24). It yields a nonlinear expression to be minimized, which can be achieved by a numerical method. We define the following function $I_c(T, \alpha_k, \sigma_k)$:

$$I_c(T, \alpha_k, \sigma_k) = \frac{1}{Z(\alpha_k, \sigma_k, p_k)} \iint_{(\xi,n)\in\mathbb{C}^2} |\theta_T(\xi + n) - \xi|^2 e^{-|n|^2/2\sigma_k^2 - |\xi|^{p_k}/\alpha_k} \, d\xi \, dn \tag{3.16}$$

Because of the dimensionality of this integral and the expression for Z, an optimized Monte Carlo [41] method is preferred to classical numerical integration (the gain in speed is about 100 times in this case). To achieve this, the complex random variables are sampled w.r.t. the prior law (3.14) and w.r.t. the Gaussian law of the noise. Then, the risk is averaged over a large number of samples (400000 in our experiment). The shape of this function is illustrated in Fig. 3.23 for different values of α_k and σ_k, for $p_k = 1$. Since it has a nearly quadratic behaviour near the optimum, the minimization w.r.t. T is made by a Newton-Raphson algorithm. The values of $\hat{T}$ computed this way can be stored in a table, to be used later within a thresholding algorithm.

Modeling by a Laplacian law ($p_k = 1$)

Fig. 3.24 shows the variation of the optimal value $\hat{T}$ with α_k and σ_k. A good approximation of this value is given by the following equation :

$$\hat{T_k} \simeq 0.75\, \sigma_k \left(\frac{\sigma_k}{\alpha_k}\right)^{0.8} \tag{3.17}$$

To compute $\hat{T}$ the value of α_k is needed. It can be estimated from a noisy subband, by computing the variances of each law. We have $x = \xi + n$ for each coefficient. We assume that signal ξ and noise n are independent :

$$E[|x|^2] = E[|\xi|^2] + E[|n|^2] \tag{3.18}$$

The expression of the law (3.14) gives $E[|\xi|^2] = 2\alpha_k^2$. Since n is a complex we have $E[|n|^2] = 2\sigma_k^2$. $E[|x|^2]$ can be estimated on the subband k by a sum of squares, n_k is the number of coefficients of this subband. So we obtain :

$$\alpha_k = \sqrt{\frac{1}{2\, n_k} \sum_i |x_i|^2 - \sigma_k^2} \tag{3.19}$$

Evaluating α_k this way can present some problems if the image size is insufficient, because the variance estimate is not accurate enough. Thus, a minimum size of 512×512 is recommended, else the estimation has to be done differently, by computing the ML for example. In practice, the prior parameters α_k can be evaluated manually and their value can be used for a large set of images having the same properties (corresponding to the same type of scene or imaging system).

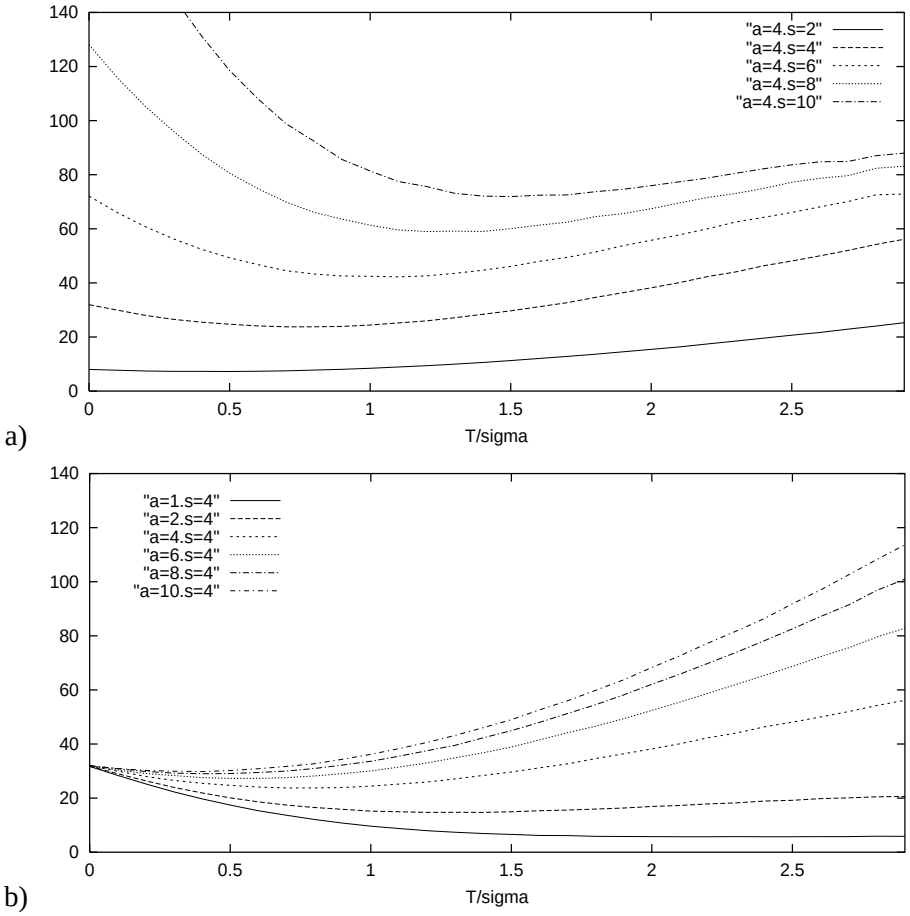


Figure 3.23: *Risk as a function of T/σ_k, computed with $p_k = 1$ (Monte Carlo method) : a) different values of σ_k with $\alpha = 4$, b) different values of α with $\sigma_k = 4$*

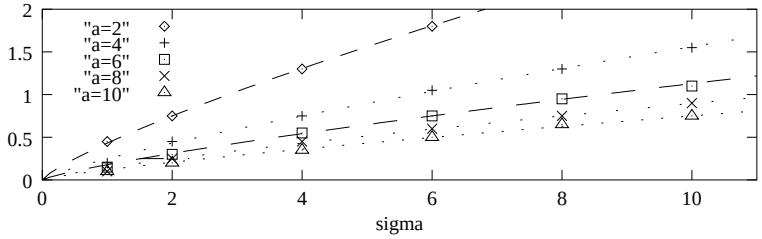


Figure 3.24: *$\hat{T}/\sigma_k$ as a function of σ_k, computed for different values of α, with $p_k = 1$ (Monte Carlo method), the lines represent the model (3.17)*

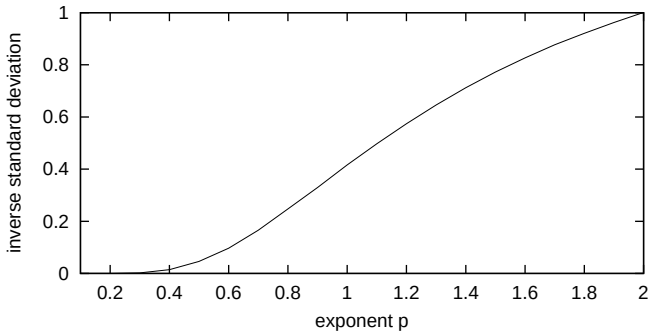


Figure 3.25: *Inverse standard deviation of the complex Generalized Gaussian law (3.14) function of p_k, with $\alpha_k = 1$*

Generalized Gaussian law with $p_k \neq 1$

To get a more accurate estimate of the optimal threshold, let us take a more general model by setting $p_k \neq 1$. The same method is used to compute the optimal threshold numerically. If we choose a fixed value of p_k, for example $p_k = 0.7$ because it seems to efficiently model the subbands of the chosen satellite image, then we obtain the approximate law :

$$\hat{T_k} \simeq 0.67\,\sigma_k \left(\frac{\sigma_k^{0.6}}{\alpha_k^{0.55}} \right) \tag{3.20}$$

Then the value of α_k is computed the same way as before. We have a more complicated expression for $E[|\xi|^2]$. We remark by using equ. (3.14) that the variance is given by $E[|\xi|^2] = F(p_k)\,\alpha_k^2$. $F(p_k)$ denotes a function of the exponent p_k, which can be computed numerically by sampling ξ for different values of p_k with $\alpha_k = 1$ and evaluating the expectation by an empirical mean. We finally obtain :

$$\alpha_k = \sqrt{\frac{1}{F(p_k)} \left(\frac{1}{N_k} \sum_i |x_i|^2 - 2\sigma_k^2 \right)} \tag{3.21}$$

So we find again an equation like (3.19), weighted by the term $F(p_k)^{-1/2}$. This term is illustrated by Fig. 3.25. If p_k is unknown, the order-4 moment of the prior law has to be estimated to provide another equation, to enable estimation of the two unknown parameters of this law.

3.3.3 Using a Bayesian homogeneous model

It is possible to estimate the unknown wavelet coefficients within a Bayesian framework, as in the real case. We have defined a prior distribution for these coefficients. To compute the MAP estimate of ξ, we use Bayes law to calculate the expression of the posterior probability :

$$\hat{\xi} = \arg\max_{\xi} P(\xi \mid x) = \arg\max_{\xi} P(x \mid \xi) P(\xi) \tag{3.22}$$

where $P(x \mid \xi)$ is given by the law of the noise, since $x = \xi + n$:

$$P(x \mid \xi) = \frac{1}{2\pi\, \sigma_k^2} e^{-|x-\xi|^2/2\sigma_k^2} \tag{3.23}$$

Then we obtain, by using the prior law (3.14), the following expression of the MAP :

$$\hat{\xi} = \arg\min_{\xi} \left[|x-\xi|^2/2\sigma_k^2 + |\xi/\alpha_k|^{p_k} \right] \tag{3.24}$$

It is possible to prove that this is equivalent to applying a thresholding operator θ^{p_k} to the magnitude of each coefficient, without modifying the phase :

$$\hat{\xi} = \theta^{p_k}(x) \tag{3.25}$$

This is shown by setting that the derivatives of the above expression, w.r.t. both real and imaginary parts of ξ, to be null. It leads to the following equation :

$$x = \xi \left(1 + p \frac{\sigma_k^2}{\alpha_k^{p_k}} |\xi|^{p_k - 2} \right) \tag{3.26}$$

which splits into two equations for magnitude and phase :

$$|x| = |\xi| \left(1 + p_k \frac{\sigma_k^2}{\alpha_k^{p_k}} |\xi|^{p_k - 2} \right) \tag{3.27}$$

$$\text{and} \quad \arg(x) = \arg(\xi) \tag{3.28}$$

This shows that ξ is obtained from x by keeping the phase and thresholding the magnitude.

This reduces to simple soft thresholding (equ. (3.10)) if $p_k = 1$. Thus, this type of attenuation naturally comes from the Bayesian approach. In this case, we have :

$$\hat{\xi} = \theta^s_{\hat{T}}(x) \quad \text{with} \quad \hat{T} = \frac{\sigma_k^2}{\alpha_k} \tag{3.29}$$

In this case the threshold $\hat{T}$ is known, and it is given by σ^2/α. This expression is approximately equal to the previously found law (3.17).

If we consider the law (3.14) with an exponent $p_k \neq 1$, we have to solve numerically equ. (3.27) w.r.t. $|\xi|$. The solution has the behaviour of a thresholding function, which we

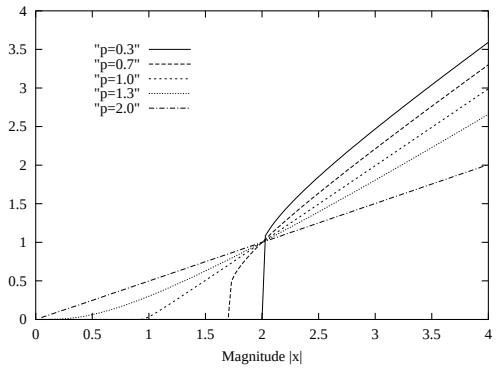


Figure 3.26: *Magnitude thresholding functions $\theta^{p_k}(|x|)$ corresponding to different values of p_k*

denote $\theta_T^{p_k}(x)$. It is a real function, which is only applied to the magnitude of the complex coefficients. Fig. 3.26 shows some of these functions for different values of p_k.

The resulting thresholding functions are smoother for $p_k > 1$, and they become linear if $p_k = 2$, as in the real case [43]. If $p_k < 1$, which seems to be more realistic (see Fig. 3.21), the functions become discontinuous.

The corresponding threshold value is found by differentiating equ. (3.27) w.r.t. $|\xi|$:

$$\hat{T} = \frac{2 - p_k}{1 - p_k} \left[p_k (1 - p_k) \frac{\sigma_k^2}{\alpha_k} \right]^{1/(2-p_k)} \tag{3.30}$$

3.3.4 Using the noninformative Jeffrey's prior

It is possible to use a different approach for subband modeling. It has been proposed in [19] within a wavelet based image denoising framework.

This approach is based on the following assumption : the inference procedure should be invariant under changes of amplitude and scale. It means that the prior probability law of ξ must keep the same behaviour even it is rescaled by setting for example $\xi' = k\xi$. Since it is not the case for classical Gaussian or Laplacian models, the author uses the following prior, which is called noninformative prior [5] :

$$P(\xi) \propto \frac{1}{|\xi|} \tag{3.31}$$

This corresponds to an extremely heavy-tailed distribution, which approximately describes the original wavelet coefficients ξ. But it is improper : the resulting posterior density function is not integrable.

Therefore, an alternative to the fully Bayesian framework is chosen. It consists of considering that the real or imaginary part of each coefficient is a zero-mean Gaussian variable, of variance s^2. This defines an adaptive (inhomogeneous) model, since each coefficient has

a different variance. Then, this variance is supposed to follow Jeffrey's hyperprior distribution, i.e. we have :

$$P(s^2) \propto \frac{1}{s^2} \tag{3.32}$$

This is equivalent to $P(\xi) \propto 1/|\xi|$ within a Bayesian context. Thus, the model finally remains homogeneous, even if an adaptive Gaussian model is used intermediately to address the problem of the improper distribution.

The estimation is then performed in two steps :

- estimate the variance $\hat{s}^2$ by using the MAP : $\hat{s}^2 = \arg\max_{s^2} P(s^2 \mid x)$
- estimate the unknown coefficient by using the MAP : $\hat{\xi} = \arg\max_{\xi} P(\xi \mid x)$

To express $P(s^2 \mid x)$, we need $P(x \mid s^2)$ and $P(s^2)$. Since both signal and noise are Gaussian, of respective variances s^2 and σ_k^2, we have $P(x \mid s^2) = P(x_r \mid s^2)P(x_i \mid s^2)$, then $P(x_r \mid s^2) = P(x_i \mid s^2) = N(N(0, s^2), \sigma_k^2) = N(0, s^2 + \sigma_k^2)$. We also have the hyperprior (3.32). Then we obtain :

$$\hat{s}^2 = \arg\max_{s^2} P(s^2 \mid x) \propto \frac{1}{(s^2 + \sigma_k^2)^2} e^{-|x|^2/2(s^2+\sigma_k^2)} \tag{3.33}$$

which is equivalent to $s^2 + \sigma_k^2 = |x|^2/4$, which gives :

$$\hat{s}^2 = \begin{cases} |x|^2/4 - \sigma_k^2 \text{ if } |x|^2 \geq 4\sigma^2 \\ 0 \text{ if } |x|^2 < 4\sigma^2 \end{cases} \tag{3.34}$$

The MAP estimate $\hat{\xi} = \arg\max_{\xi} P(\xi \mid x)$ for the inhomogeneous Gaussian model gives the following expression (see next section for a complete proof) :

$$\hat{\xi} = \frac{s^2}{s^2 + \sigma_k^2}\, x \tag{3.35}$$

By combining equations (3.34) and (3.35) we obtain the following estimate, which we call noninformative thresholding function θ_J :

$$\hat{\xi} = \theta_J(x) = \frac{|x|^2 - 4\sigma_k^2}{|x|^2}\, x \text{ if } |x|^2 \geq 4\sigma^2, \quad 0 \text{ elsewhere} \tag{3.36}$$

The advantage of such a method is that there is no need for parameter estimation, since there is no parameter. It gives acceptable deconvolution results, as illustrated by figure 3.27. The model captures the heavy-tailed distribution of the coefficients, and the computation is made possible by this two-step approach, in spite of the non-integrability of the density function.

However, we can notice a drawback of this method, which is especially visible in homogeneous areas : the residual noise is quite visible, compared to the previously seen techniques (see Fig. D.1, appendix D). This probably comes from the lack of robustness of the estimation method. We can remark that if we remove the prior law of s^2 (flat hyperprior), the estimation of s^2 is done by the MLE, which gives a function like (3.34) but with a threshold $2\sigma^2$ instead of $4\sigma^2$, which is insufficient since the magnitude of the noise has a variance equal to $2\sigma^2$. Thus, the hyperprior makes the estimation more robust, by doubling the threshold value. It is still not sufficient to efficiently remove noise peaks in large constant areas. Hereafter, we will detail a more robust method to filter the complex wavelet packet subbands.

3.3.5 Using a Bayesian adaptive Gaussian model

The inhomogeneous model

The Generalized Gaussian models presented previously have been chosen because they seem to correctly match the original coefficients distribution, which is heavy-tailed. Another possible way to capture this property is to define an inhomogeneous model, which adapts to the local characteristics of the subbands. Some approaches to spatial adaptivity in the real wavelet domain can be found in the literature, see for example [4, 8].

To simplify, let us choose a Gaussian model. The variance parameter is different for each coefficient, which enables to differentiate edges or textures, which have a high intensity, from the homogeneous areas, which generally correspond to very low values of the coefficients.

Since the parameters can be very different from one variable to another one, the histogram of a subband can have a heavy-tailed behaviour, even if the distribution is Gaussian.

We denote s^2 the variance of the real or imaginary part of an original coefficient ξ, as previously. We have the prior law :

$$P(\xi \mid s) = \frac{1}{2\pi\, s^2} e^{-|\xi|^2/2s^2} \tag{3.37}$$

If the parameters of the prior distribution are known, the unknown variables are estimated by computing the MAP. It is a fully Bayesian technique, we use the property $P(\xi \mid x) \propto P(x \mid \xi)P(\xi)$. Recall the expression of the noise distribution (3.23), and combine it with equ. (3.37) to obtain :

$$P(\xi \mid x) \propto e^{-|x-\xi|^2/2\sigma_k^2 - |\xi|^2/2s^2} \tag{3.38}$$

Therefore, the MAP is given by :

$$\hat{\xi} = \arg\min \left[\frac{|x-\xi|^2}{2\sigma_k^2} + \frac{|\xi|^2}{2s^2} \right] \tag{3.39}$$

To compute the minimum, differentiate w.r.t. the real and the imaginary parts of ξ, which gives two equations, which are recombined to form an equation with complex numbers :

$$\begin{cases} \frac{\xi_r + x_r}{\sigma_k^2} + \frac{\xi_r}{s^2} = 0 \\ \frac{\xi_i + x_i}{\sigma_k^2} + \frac{\xi_i}{s^2} = 0 \end{cases} \quad \Leftrightarrow \quad \frac{\xi + x}{\sigma_k^2} + \frac{\xi}{s^2} = 0 \tag{3.40}$$

which finally gives the inhomogeneous MAP estimate :

$$\hat{\xi} = \frac{s^2}{s^2 + \sigma_k^2}\, x \tag{3.41}$$

Adaptive parameter estimation

The most difficult problem in this approach is to estimate the adaptive parameters of the model. As it is shown in [28], the MLE is not robust when applied on incomplete data (i.e. when the estimation is made on noisy data). But the robustness becomes acceptable when the estimation is made on an image which is a good approximation of the unknown image $\mathcal{X}$. Here, we will use the same approach to estimate the variances s^2.

Consider equ. (3.37). The complete data MLE is defined by $\hat{s}^2 = \arg\max_{s^2} P(\xi \mid s^2)$. By independence assumptions, we obtain :

$$\hat{s}^2 = \frac{|\xi|^2}{2} \tag{3.42}$$

where the factor 2 is coming from the dimensionality of the distribution.

We obviously do not have access to the original coefficients, that is why we take the transform coefficients of an approximate original image instead. Experiments have shown that a satisfactory approximation is provided by a nonlinear regularizing algorithm, such as RHEA, detailed in [25]. It essentially consists of a variational method (minimization of a criterion which penalizes noisy solutions, but preserves edges) preceded by an automatic parameter estimation step to compute the hyperparameters of the regularizing model.

This method is certainly not perfect, and some residual noise remains, it is visible in constant areas (see Fig. 4.11). However, we filter this noise as well as the deconvolved noise, by a thresolding technique. We choose to use the noninformative threshold of the previous section, because it does not require any additional parameter estimation.

The proposed algorithm consists of obtaining the desired approximate original image using RHEA [25], filtering the CWPT of the result using equ. (3.36), estimating the adaptive parameters using the complete data MLE with equ. (3.42), and then estimating the unknown coefficients by computing the MAP by equ. (3.41).

In addition to the computation of the deconvolved noise variance, we also need to compute the variance $2\tilde{\sigma}_k^2$ of the residual noise of the approximate image, to correctly apply equ. (3.36) (see section 3.3.7 for the expressions of σ_k^2 and $\tilde{\sigma}_k^2$).

We denote $\tilde{\xi}$ the thresholded transform coefficients of the approximate image. It is supposed to be sufficiently exempt from noise, and to content sufficient information to enable

texture and edge recovery. Homogeneous areas and edges are fine, since we have used an edge-preserving method followed by an efficient noise thresholding. But it remains to prove why the coefficients related to textured areas are sufficiently high to avoid a too strong attenuation by using equ. (3.41) in these areas. It seems that the used variational method does not completely remove the textures, and even if visually they are not very sharp, they are sufficiently present in the approximate image to enable a correct reconstruction by the method detailed here.

Finally, if $\tilde{\xi}$ is known, by using equations (3.41) and (3.42), the expression of the coefficient estimate is :

$$\hat{\xi} = \frac{|\tilde{\xi}|^2}{|\tilde{\xi}|^2 + 2\sigma_k^2}\, x \tag{3.43}$$

Comparison with oracle threshold

Let us recall the expression of the oracle threshold (3.9), i.e. the expression of the thresholding function which minimizes the maximum risk of the estimator $\hat{\xi}$:

$$\theta_o(x) = \frac{|\xi|^2}{|\xi|^2 + 2\sigma_k^2}\, x \tag{3.44}$$

It has exactly the same expression as equ. (3.43), if we take the approximate original coefficient $\tilde{\xi}$ instead of ξ. Both Bayesian and minimum risk methods lead to the same expression, which means that the chosen model is good, since the corresponding estimator provides the minimum risk.

Like in the case of Wiener filtering [23], computing the MAP under Gaussian assumptions (both signal and noise have Gaussian distributions) is equivalent to minimize the risk of a linear estimator w.r.t. the attenuation factor. Therefore, the two approaches are equivalent in the Gaussian case.

3.3.6 Comparison of the different estimation schemes

Comparison of the results

Fig. 3.27 shows the results of the different thresholding methods previously presented. We essentially retain 4 different methods, since we have $p_k = 1$ or $p_k = 0.7$, and we can either use a minimum risk computation (see section 3.3.1) or a Bayesian filtering technique (see section 3.3.3) to optimally threshold the transform coefficients.

The more accurate is the model, the better is the visual quality of the results, as illustrated by the difference between right and left pictures. That is why we prefer using $p_k = 0.7$, which is closer to reality than the Laplacian law, if we process aerial or satellite images.

The homogeneous Bayesian approach gives better *visual* results than the minimum risk computation, this difference can be seen between top and bottom pictures of Fig. 3.27. The

former naturally introduces thresholding functions which are optimal w.r.t. the value of the exponent p_k, while the latter imposes a given thresholding function and tries to optimize the SNR. In fact, having the highest SNR does not necessary mean that the restoration quality is optimal. Then, we prefer to use the Bayesian approach, since we only impose a prior model, whereas computing the minimum risk requires to impose a fixed thresholding function and to define a quality criterion.

With regard to the adaptive model, the results are visually much better, and the SNR is slightly higher. Details are better preserved and constant areas are cleaner. That is why we retain this model, in the final version of the algorithm (3.4.2). The corresponding results are presented on Fig. 4.13, in comparison with other existing techniques.

If we try to compare the Bayesian and the minimum risk approaches, we find that the SNR generally differs from less than $0.1\,dB$ if we apply equ. (3.44) instead of (3.43), and the visual comparison is difficult. The Bayesian approach gives less noisy results, while the other approach could preserve a few more details - we choose the Bayesian point of view in the final algorithm, but the other perspective could be used.

3.3.7 Noise variance estimation

The variance of the deconvolved noise in each subband k is needed to estimate the optimal threshold. Since the wavelet packet transform is linear, the noise remains Gaussian. As for the real wavelet packet transform, the noise variance can simply be computed by considering the transform as the result of a decimated convolution.

For each subband k, the convolution kernel (or the impulse response) R^k of the real or imaginary part is obtained by expressing a vector of the transform basis $\mathcal{B}$ in the original image basis. This can be simply achieved by setting to 1 or i the corresponding transform coefficient, and the other ones to 0, and then processing the reverse transform. R^k can either correspond to the real or the imaginary part of the complex transform. Indeed, as seen in equ. (3.7), real and imaginary parts are independent because the law is Gaussian.

To estimate the noise variance, the computation is done in the Fourier basis because the convolution operators related to R^k and h are diagonal in this basis. See appendix C for the details of this computation.

Deconvolution without regularization

Since the CWPT of the deconvolved noise without regularization, denoted N_d, is given by :

$$\mathcal{F}[N_d] = \frac{\mathcal{F}[N]}{\mathcal{F}[h]} \tag{3.45}$$

The variance $2\sigma_k^2$ in the subband k is given by :

$$\sigma_k^2 \simeq \sigma^2 \, \frac{1}{N_x N_y} \sum_{ij} \frac{|\mathcal{F}[R^k]_{ij}|^2}{|\mathcal{F}[h]_{ij}|^2} \tag{3.46}$$

Minimum risk, p_k=1
Soft thresholding, equ. (3.17)

SNR = 23.8 dB

Minimum risk, p_k=0.7
Soft thresholding, equ. (3.20)

SNR = 23.7 dB

Bayesian approach, p_k=1
Soft thresholding, equ. (3.29)
SNR = 23.4 dB

Bayesian approach, p_k=0.7
Bayesian thresholding function
SNR = 23.3 dB (Fig. 2.6)

Figure 3.27: *Deconvolution results on image of Fig. 1.1 with different homogeneous parametric thresholding methods*

Deconvolution with quadratic regularization

The CWPT of the deconvolved noise with a quadratic regularization governed by a parameter b, denoted N_b, is given by :

$$\mathcal{F}[N_b] = \frac{\mathcal{F}[N]\mathcal{F}[h]^*}{|\mathcal{F}[h]|^2 + 2\sigma^2 b(|\mathcal{F}[d_x]|^2 + |\mathcal{F}[d_y]|^2)} \tag{3.47}$$

where d_x and d_y are the convolution kernels corresponding to the first order derivatives w.r.t. columns and rows. The variance $2\tilde{\sigma}_k^2$ in the subband k is then given by :

$$\tilde{\sigma}_k^2 \simeq \sigma^2 \frac{1}{N_x N_y} \sum_{ij} \frac{|\mathcal{F}[R^k]_{ij}|^2 |\mathcal{F}[h]_{ij}|^2}{\left(|\mathcal{F}[h]|_{ij}^2 + 2\sigma^2 b(|\mathcal{F}[d_x]|_{ij}^2 + |\mathcal{F}[d_y]|_{ij}^2)\right)^2} \tag{3.48}$$

This expression can also be used to compute the variance of the noise remaining after an edge-preserving regularization, based on a φ function [21, 25]. Such methods provide a quadratic regularization in constant areas, because they only contain noise. The magnitude of the pixel differences is low, so it is located in the quadratic part of the φ function. Thus, in these areas, the remaining noise after a regularized deconvolution is the same for quadratic and non-quadratic φ functions. Therefore, we know the variance of the noise which contaminates the "ground truth" image used in the method described in Section 3.3.5.

3.4 The proposed algorithm

3.4.1 The COWPATH algorithm

The proposed deconvolution method, called COWPATH (COmplex Wavelet Packet Automatic THresholding), exists in 2 versions. Version 1 is based on an homogeneous modeling of the subbands, and we essentially retain the 2 methods giving the best results (Generalized Gaussian and noninformative prior, see sections 3.3.5 and 3.3.4). Version 2 is based on an adaptive Gaussian modeling of the subbands, and the parameter estimation needs to first deconvolve the image with the RHEA [25] algorithm.

ALGORITHM 3.4.1 (COWPATH 1)

- *DCT (discrete cosine transform) of the observation Y*
- *Deconvolution : divide by $\mathcal{F}[h]$ (in practice, divide by $\mathcal{F}[h]+\epsilon$ where ϵ is small, since some of the coeffi cients can be null)*
- *Inverse DCT of the result, which gives X*
- *CWPT (complex wavelet packet transform) of X (see Algo. 3.2.1)*
- *Computation of σ_k using the known h and σ (see equ. (3.46))*

- *Coeffi cient thresholding, using one of the following methods :*
  1. *Either, estimation of α_k from CWPT[X] and σ_k (see equ. (3.21))*
     Coeffi cient thresholding using numerically computed functions (see Fig. 3.26)
  2. *Or, coeffi cient thresholding using a noninformative prior (see equ. (3.36))*
     (no parameter to estimate)
- *Inverse CWPT, which gives the estimate $\hat{X}$.*

ALGORITHM 3.4.2 (COWPATH 2)

- *DCT (discrete cosine transform) of the observation Y*
- *Deconvolution : divide by $\mathcal{F}[h]$ (in practice, divide by $\mathcal{F}[h]+\epsilon$ where ϵ is small, since some of the coeffi cients can be null)*
- *Inverse DCT of the result, which gives X*
- *CWPT (complex wavelet packet transform) of X (see Algo. 3.2.1)*
- *Computation of the approximate original image $\tilde{X}$: apply the RHEA algorithm [25] on Y (nonlinear regularization, with automatic parameter estimation)*
- *CWPT (complex wavelet packet transform) of $\tilde{X}$ (see Algo. 3.2.1)*
- *Computation of σ_k using the known h and σ (see equ. (3.46))*
- *Computation of $\tilde{\sigma}_k$ (residual noise on approximate original image) using the known h and σ (see equ. (3.48))*
- *Thresholding of the approximate image coeffi cients using a noninformative prior and $\tilde{\sigma}_k$ (see equ. (3.36))*
- *Estimation of the parameters s of the inhomogeneous Gaussian model (see equ. (3.42))*
- *Coeffi cient thresholding by computing the MAP (see equ. (3.41))*
- *Inverse CWPT, which gives the estimate $\hat{X}$.*

It is possible to replace the nonquadratic regularizing model of the RHEA algorithm by a simple quadratic model. It produces results equivalent to Wiener [23] filtering. The edges are not as sharp as with the nonquadratic model, but this approximate image is sufficiently accurate to provide a correct estimation of the inhomogeneous parameters of the subband model (the SNR difference between the original and accelerated algorithm is about 0.1 dB for the SPOT 5 simulation image presented in Chapter 4).

ALGORITHM 3.4.3 (COWPATH 2 - ACCELERATED VERSION)
Replace the RHEA algorithm [25] by a quadratic regularization method :

- *Parameter estimation using the MLE (fast deterministic algorithm, consisting in computing sums in the DCT domain, see [28])*
- *One-step deconvolution in the DCT domain, using the estimated parameter value*
- *Inverse DCT of the result, which gives* $\tilde{X}$

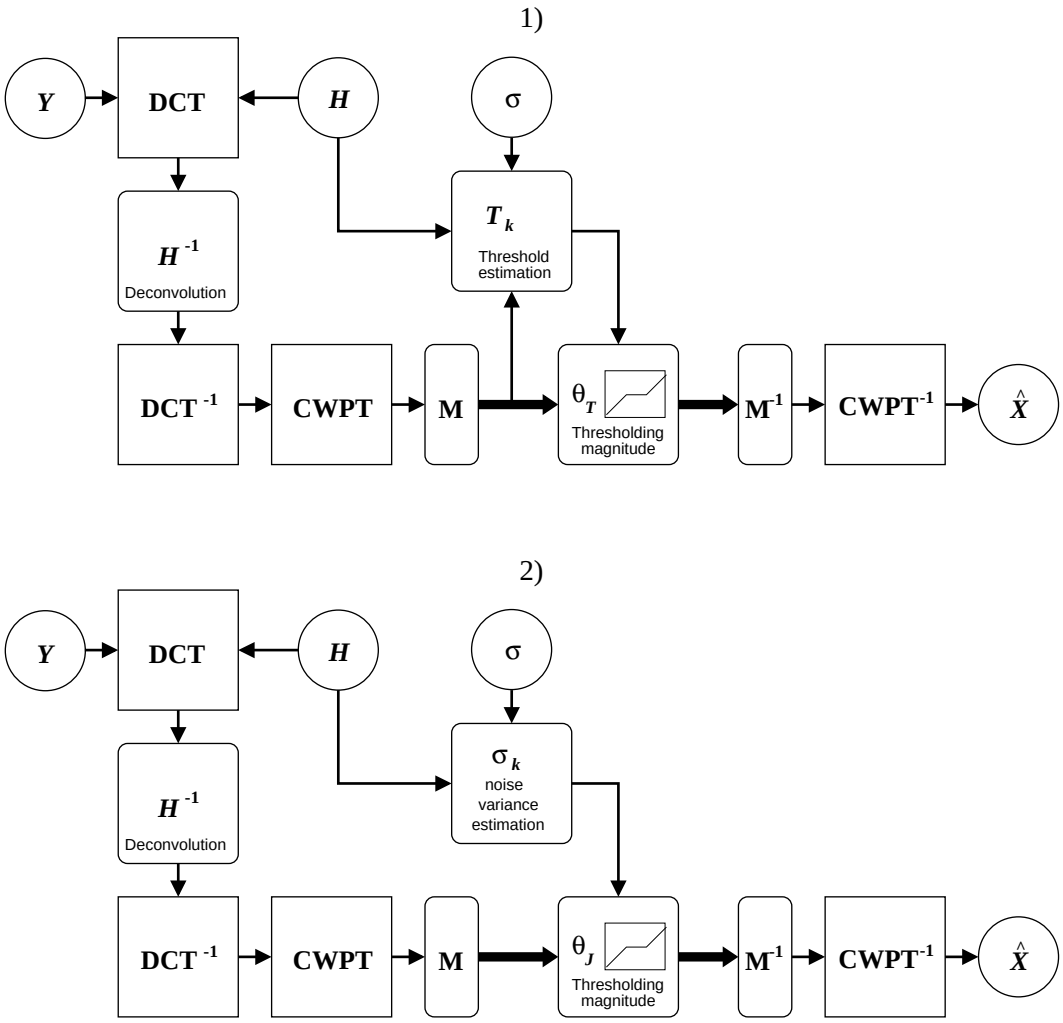


Figure 3.28: *The COWPATH 1 algorithm methods 1 and 2 (COmplex Wavelet Packet Automatic THresholding) (see Fig. 3.6 for the matrix M)*

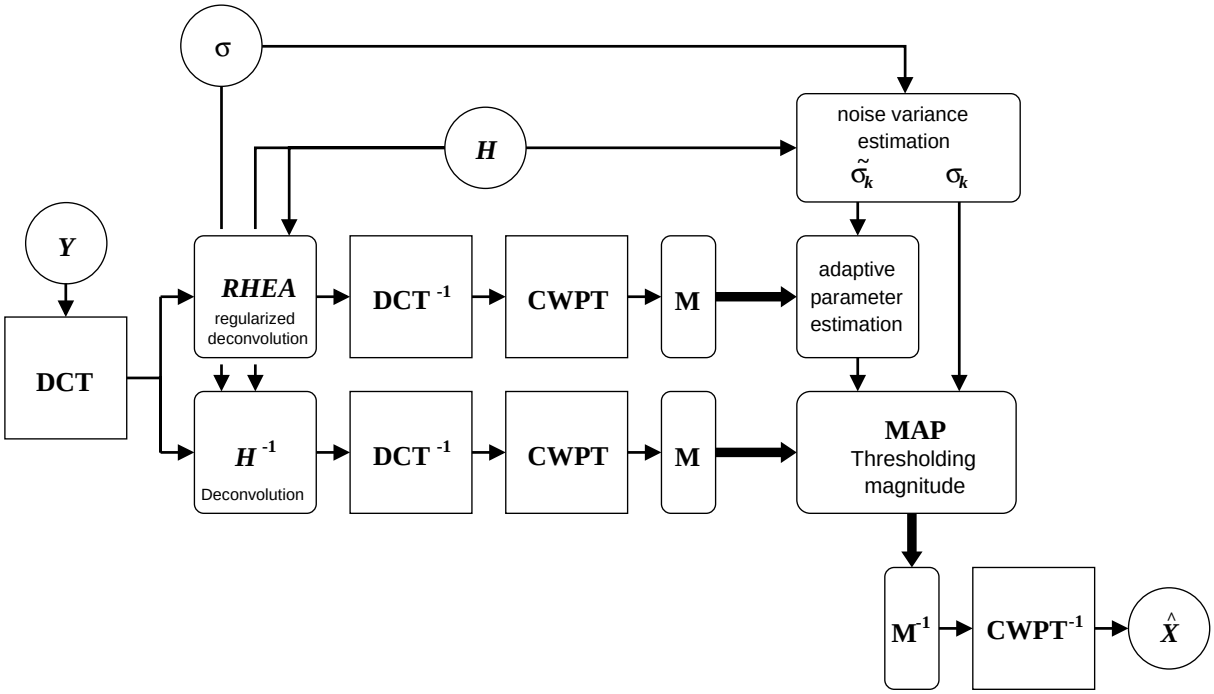


Figure 3.29: *The COWPATH 2 algorithm (COmplex Wavelet Packet Automatic THresholding) (see Fig. 3.6 for the matrix M)*

Figure 3.30: *Result of the COWPATH 2 algorithm on image of fig. 1.1*

The use of a DCT instead of a FFT allows us to fulfill the symmetric boundary conditions, so we avoid producing artefacts on the borders of the image [24, 25, 27]. This is equivalent to using images 4 times larger, symmetrically extended w.r.t. rows and columns, but we process real pixels, unlike the original algorithm. Due to the symmetry in the frequency domain, the number of processed pixels remains constant $(N_x N_y)$.

In practice, for the image provided by the French Space Agency (CNES), only 14 subbands are needed for the filtering algorithm. The highest frequencies are set to 0 because the highest coefficient in these bands is lower than σ_k, which means that these subbands only contain noise.

Fig. 3.30 shows the result of the proposed algorithm. The results for the whole image are presented in Chapter 4. To improve the SNR, the subbands of the wavelet transform $z_{\pm}^{11}, z_{\pm}^{12}, z_{\pm}^{13}$ of the result could be filtered by hard thresholding, with $T_k = 3\sigma_k^r$, where σ_k^r is the standard deviation of the residual noise (as suggested in [29] for a real transform).

3.4.2 Evaluation of the residual noise

To evaluate the residual noise in homogeneous areas, after thresholding, we assume the noise is Gaussian, with known variance (determined by (3.46)). This remaining noise strongly depends on the method used to deconvolve the image.

In appendix D, we explain how to compute the residual noise variance related to the three methods giving the best results when applied on satellite images. We also give an illustration of this noise (see Fig. D.1).

The main result concerns the best algorithm (COWPATH 2). The more decorrelated the noise in the approximate image and in the deconvolved image, the lower the residual noise variance, and the resulting image improves. This is shown by equ. (D.17), where β is a decreasing function of the correlation c_k between the two noises, which multiplies a positive term.

3.4.3 Cost of the algorithm

In this computation, we neglect the cost of the DCT of the convolution kernel h (its size is small compared to the size of the entire image, and the transform may be interpolated to be resized to the size of the image). The computation of the noise variances and covariance is also neglected, since it needs the FFT of h, and of the impulse response of the complex wavelet packet subbands, which are supposed to be precomputed.

The parameter estimation step of COWPATH 1 method 1 is also neglected (it can be performed by a simple variance estimation). This is also the case for COWPATH 2.

Deconvolution

One cosine transform (DCT) needs a FFT with some adjustments, which represents about $5 \log_2 n + 15$ op/pix (operations per pixel), where n is the number of lines or columns of

a square image. The algorithm COWPATH 1 needs only 1 direct DCT and 1 inverse DCT, while COWPATH 2 needs 1 direct and 2 inverse DCTs.

Then we need for the two algorithms respectively $10 \log_2 n + 30$ and $15 \log_2 n + 45$ op/pix (120 and 180 op/pix for a 512×512 image).

Complex wavelet packet transform

The complex wavelet packet transform is efficiently implemented by a ladder filter structure [34]. The 13-19 tap odd filters h^o, g^o need 10 operations per input sample, and the 12-16 tap even filters h^e, g^e need 12 operations. The total number of operations per pixel needed to compute the transform depends on the depth of the tree. We remark that if all of the packet subbands resulting from splitting a wavelet subband are set to zero, such a splitting is not necessary (this is the case for d^{13}), and is not computed.

The first level of the transform needs 60 op/pix, and the second level 24, but it is repeated 3 times since we divide only 3 of the 4 detail subbands. It represents 132 op/pix for the complete transform. In COWPATH 1 we need 1 forward and 1 reverse transform, while in COWPATH 2 we need 2 forward transforms. It represents respectively a total of about 265 and 400 op/pix.

Case of COWPATH 2

In the case of this algorithm, we have to add the RHEA [25] step, which represents $500 + 100 \log_2 n$ op/pix (we suppose a maximum of 10 iterations, and the parameter estimation step is negligible, since it can be done by a fast method as explained in [28]).

The accelerated version (Algo. 3.4.3) uses a quadratic regularization, and the corresponding cost is negligible.

TOTAL COST

- COWPATH 1 : $10 \log_2 n + 295$ op/pix (385 op/pix for a 512×512 image)
- COWPATH 2 : $115 \log_2 n + 945$ op/pix (1980 op/pix for a 512×512 image)
- Accelerated COWPATH 2 : $15 \log_2 n + 445$ op/pix (580 op/pix for a 512×512 image)

3.5 Comparison with real wavelets

3.5.1 Directional selectivity

The impulse responses associated to the real wavelet packet transform used in algorithm 2.3.1 are plotted in Fig. 3.31. In comparison with the complex impulse responses of Fig. 3.17, it is clear that real separable packets are unable to define more than two directions. A comparable directionality can only be reached by using non-separable wavelets, or a Gabor transform. Such tools have been widely used for various applications (detection, texture segmentation) but are time consuming.

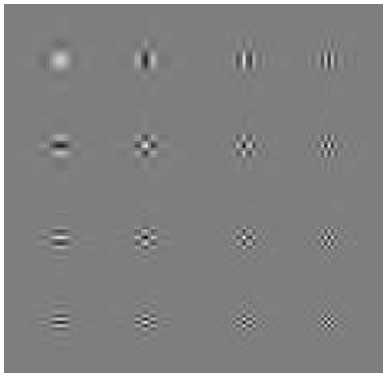


Figure 3.31: *Impulse responses of the real wavelet packet transform (Symmlet-6)*

3.5.2 Computational cost

The deconvolution process is the same for algorithms 2.3.1, 3.4.1 and 3.4.2. The difference comes from the decomposition algorithm. The tree used to compute real wavelet packets is described by Table 2.1, and we do not split the highest frequency subband of level 1, because it is set to zero. Thus, the transform needs $4l$ op/pix at level 1, where l is the size of the filter. For a Symmlet-4 wavelet (chosen because of its good frequency selectivity for a small number of coefficients) it means 32. Then, 8 op/pix are needed to perform the second level for each subband of the first level, and it is repeated 3 times due to the structure of the tree. This gives 56 op/pix for one transform. It has to be multiplied by 8 because the results have to be averaged over minimum 8 integer translations (4 pixels in each direction, which implies 16 translations, but 8 are sufficient) to avoid the artefacts coming from the lack of shift invariance.

One finally needs about 450 op/pix per pixel for the transforms, this is more than 3 times longer than the complex algorithm.

Kalifa [29] uses a different implementation, by considering a non-decimated transform, and a hybrid tree. Thus, he obtains 320 op/pixel for one wavelet packet transform. But it remains more than 2 times longer than the complex transform.

3.6 Conclusion

If we consider shift invariant methods, not only is the complex wavelet packet transform faster to compute, but it also provides a better representation of oriented features because of its directional selectivity. Kingsbury [33, 34] has shown that the complex transform is more efficient (concerning both speed and SNR) than the real transform in image denoising. In this chapter, we have proven that this is also true for image deconvolution.

This transform exhibits better directional and shift invariance properties than real wavelet packet transforms, for a lower computational cost. The proposed deconvolution method is superior to other competing algorithms, since it is faster, more accurate and fully automatic.

Chapter 4

Results

4.1 Observed data and resulting images

Quality criterion

We simply use classical Signal to Noise Ratio (SNR) computations, consisting in dividing the variance of the original image $\mathcal{X}$ by the variance of the error image (difference between the reconstructed and the original image).

$$\text{SNR}(X) = 10 \log \frac{\mathcal{X}_{ij}^2}{(\mathcal{X}_{ij} - X_{ij})^2} \tag{4.1}$$

Observed data and resulting images

- Fig. 4.1 shows the observed image, provided by the French Space Agency (CNES). It is a SPOT 5 simulation at 2.5m resolution, simulated by a convolution with the kernel h (see Fig. 1.1), and by adding a white noise, approximated by a Gaussian white noise with a standard deviation $\sigma \simeq 1.35$. The resulting SNR is 16.1 dB.
- Fig. 4.2 shows the restored image, processed with algorithm 3.4.1 (COWPATH 1), method 1 (homogeneous generalized Gaussian model). The resulting SNR is 21.3 dB.
- Fig. 4.3 shows the restored image, processed with Algo. 3.4.1 (COWPATH 1), method 2 (noninformative prior). The resulting SNR is 21.8 dB.
- Fig. 4.4 shows the restored image, processed with Algo. 3.4.2 (COWPATH 2). The resulting SNR is 22.2 dB. See Fig. 4.7 for the error image.
- Fig. 4.5 shows the restored image, processed with the accelerated version of Algo. 3.4.2 (COWPATH 1) (Algo. 3.4.3). The resulting SNR is 22.1 dB.
- Fig. 4.6 shows the original image, provided by the French Space Agency (CNES).
- Another example is given on a SPOT 3 image at 10 m resolution with the same kernel h and the same standard deviation as above. Results are shown on Fig. 4.8 and 4.9.

Figure 4.1: *Observed image of Nîmes (see Fig. 1.1 for h),* 512×512*, 256 grey levels ©CNES - SNR=16.1 dB*

Figure 4.2: *Image of Fig. 4.1 deconvolved with algorithm 3.4.1 (COWPATH 1, method 1) - SNR=21.3 dB*

Figure 4.3: *Image of Fig. 4.1 deconvolved with algorithm 3.4.1 (COWPATH 1, method 2) - SNR=21.8 dB*

Figure 4.4: *Image of Fig. 4.1 deconvolved with algorithm 3.4.2 (COWPATH 2) - SNR=22.2 dB*

Figure 4.5: *Image of Fig. 4.1 deconvolved with the fast version of Algo. 3.4.2 (COWPATH 2) - SNR=22.1 dB*

Figure 4.6: *Original image of Nîmes,* 512×512*, 256 grey levels ©CNES*

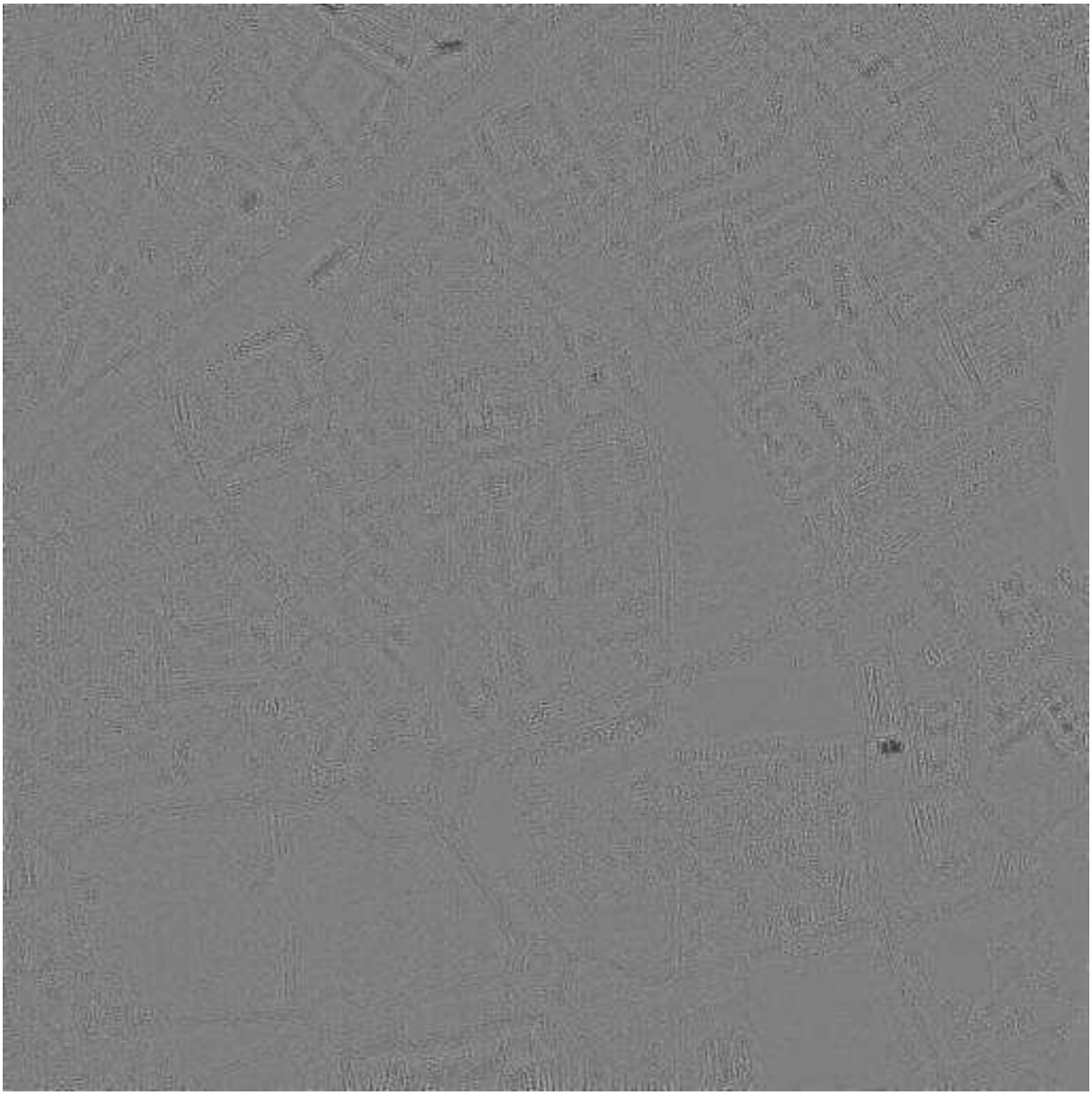


Figure 4.7: *Error image : difference between images of Fig. 4.4 and Fig. 4.6, multiplied by 2*

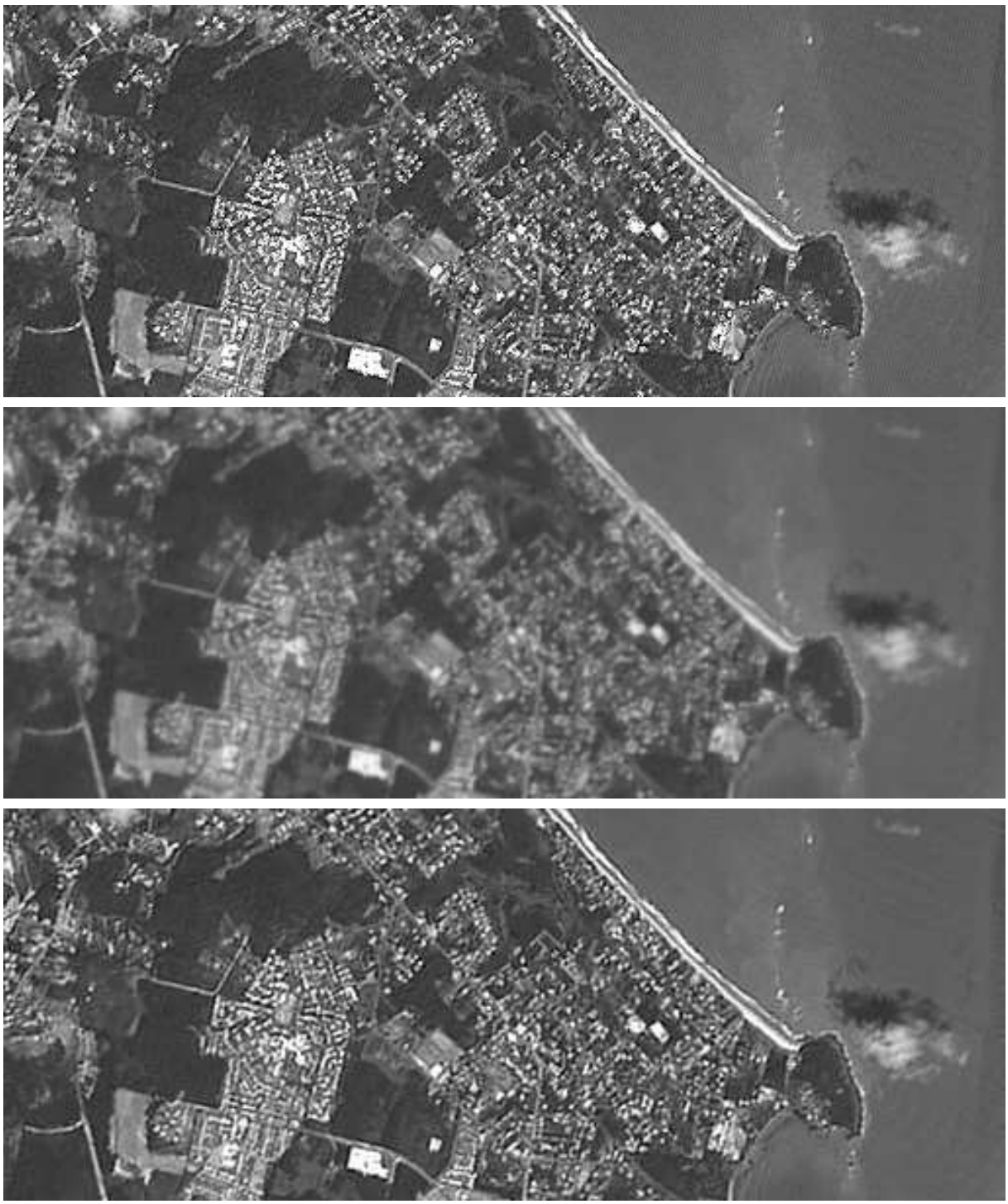


Figure 4.8: *512×200 area extracted from Cayenne (SPOT 3, resolution 10m, courtesy of Univ. of Marne-la-Vallée). Up : original, middle : blurred and noisy (same h and σ as image of Fig. 4.1), bottom : deconvolved with algorithm 3.4.2 (COWPATH 2)*

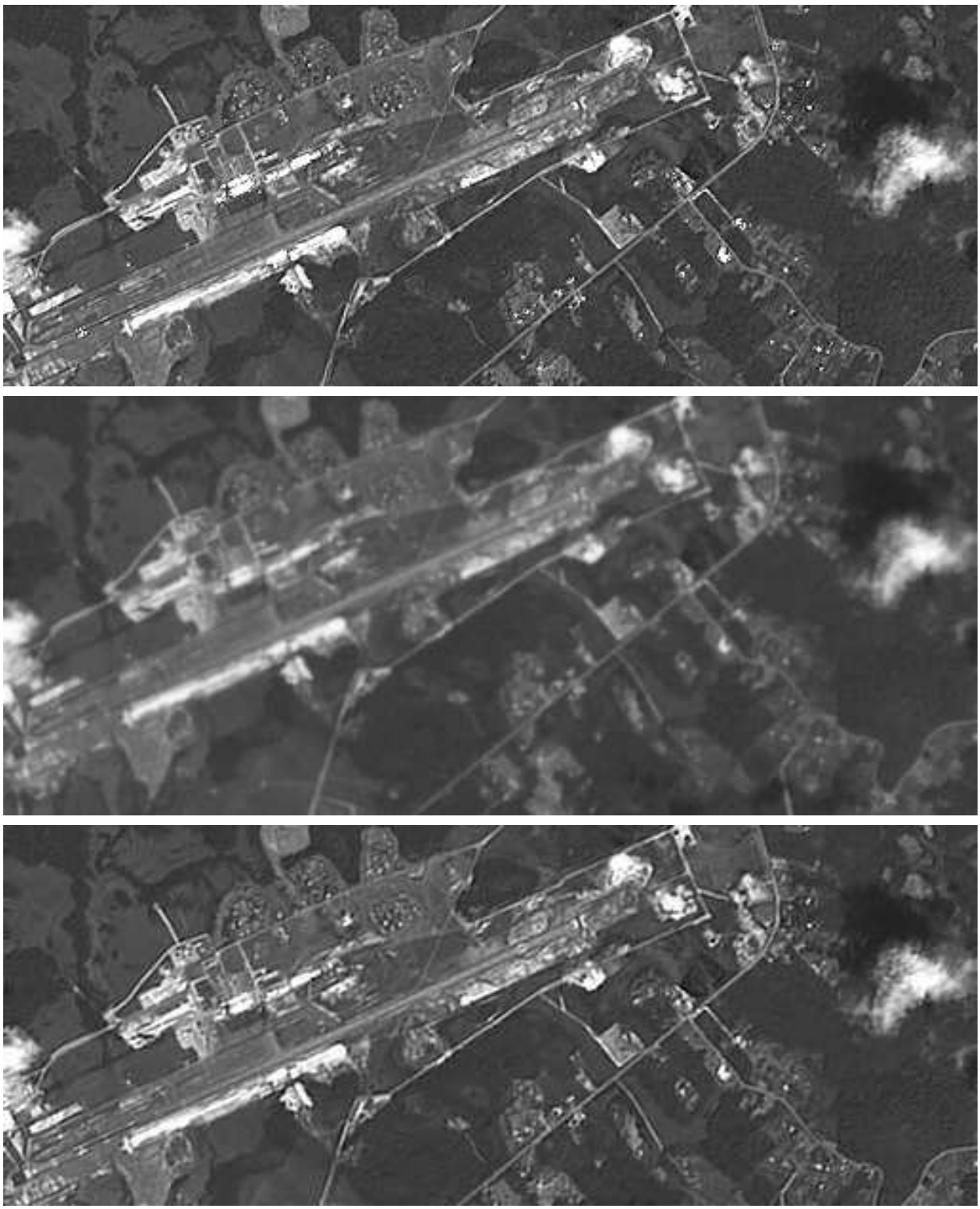


Figure 4.9: *512×200 area extracted from Cayenne (SPOT 3, resolution 10m, courtesy of Univ. of Marne-la-Vallée). Up : original, middle : blurred and noisy (same h and σ as image of fig. 4.1), bottom : deconvolved with algorithm 3.4.2 (COWPATH 2)*

4.2 Comparison with other methods

4.2.1 Quadratic regularization or parametric Wiener filter

It is possible to use a regularization method involving a quadratic function [44], consisting of minimizing the following criterion to find the estimate $\hat{X}$:

$$\hat{X} = \arg\inf_X \ \|Y - HX\|^2/2\sigma^2 + b\left(\|D_x X\|^2 + \|D_y X\|^2\right) \tag{4.2}$$

where D_x and D_y represent derivative operators w.r.t. columns and lines, and b is the regularizing parameter. We propose in [28] a method to estimate the optimal value of b. By using this method for the image of Fig. 4.1 we obtain $b \simeq 0.005$. The computation of $\hat{X}$ is easily achieved in the Fourier domain by the expression :

$$\mathcal{F}[\hat{X}] = \frac{\mathcal{F}[h]^* \mathcal{F}[Y]}{|\mathcal{F}[h]|^2 + 2\sigma^2\, b\,(|\mathcal{F}[d_x]|^2 + |\mathcal{F}[d_y]|^2)} \tag{4.3}$$

where d_x and d_y are the generators of the block circulant operators D_x and D_y.

This is nearly equivalent to the parametric Wiener filter [23], which gives the same results. It is also equivalent to isotropic diffusion [42]. The edges are filtered as well as the noise, as seen on Fig. 4.10, therefore it is impossible to obtain sharp details and noisefree homogeneous areas at the same time. Thus, the SNR remains low (about 19.7 dB) because of the insufficient noise filtering in these areas.

4.2.2 The RHEA algorithm

The RHEA algorithm has been presented in [25, 26]. It consists of minimizing the following non-quadratic criterion :

$$\hat{X} = \arg\inf_X \ \|Y - HX\|^2/2\sigma^2 + \lambda^2 \sum_{i,j} \left[\varphi\left((D_x X)_{ij}/\delta\right) + \varphi\left((D_y X)_{ij}/\delta\right)\right] \tag{4.4}$$

where φ is a non-quadratic function, whose behaviour enables to preserve the edges while filtering the noise only along these edges and in constant areas. The parameters λ and δ are estimated automatically. The resulting image is represented on Fig. 4.11 and exhibits sharp edges, compared to the previous one. However, some noise remains in homogeneous regions, and textures are attenuated.

4.2.3 Real wavelet packet thresholding

Using Algo. 2.3.1 enables to recover the textures, and to obtain very smooth homogeneous areas, which was not the case with the two previous methods (see Fig. 4.12). However, the edges are less sharp than with the RHEA algorithm.

The diagonal features are better restored with the COWPATH algorithm, thanks to its rotational invariance. That explains the difference between the visual quality of real and complex methods. Furthermore, the real method is a bit slower than the complex one, due to the means used to ensure the shift invariance.

Figure 4.10: *Image of Fig. 4.1 deconvolved with quadratic regularization (Tikhonov) and optimal hyperparameter* $b = 0.0005$ *- SNR=19.6 dB*

Figure 4.11: *Image of Fig. 4.1 deconvolved with RHEA algorithm - SNR=22.0 dB*

Figure 4.12: *Image of Fig. 4.1 deconvolved with algorithm 2.3.1 (real wavelet packets) - SNR=21.8 dB*

Original ($\mathcal{X}$) Observed (Y) Quadratic regularization

RHEA COWPATH 1, method 1 COWPATH 1, method 2

Real wavelet packets COWPATH 2 accelerated COWPATH 2

Figure 4.13: *Zoom on a* 128×128 *area extracted from the image of Fig. 4.1: comparison of the different deconvolution methods (see previous figures for entire images)*

Chapter 5

Conclusion and future work

5.1 Conclusion

We have proposed a deconvolution method, consisting of thresholding the image deconvolved without regularization in a complex wavelet packet basis. This method is more accurate and faster than wavelet-based competing algorithms and it is also automatic.

However, the proposed algorithm, like all other methods involving a wavelet representation, is unable to perfectly restore the sharp edges, because the wavelet basis vectors have a poor spatial localization compared to these details. That is why we prefer to use the result of the COWPATH algorithm not as is, but as a starting point of an inhomogeneous parameter estimation method, like the complete data MLE method described in [28]. We show in [28] that the result of the COWPATH algorithm is sufficiently accurate to provide correct parameter values to be injected into an adaptive regularizing model, because the texture and homogeneous areas are correctly restored and because the used estimator is robust w.r.t. the blur which affects the edges.

5.2 Further developments

It is possible to extend the proposed approach to take into account various levels of noise and different convolution kernels.

Furthermore, the nearly diagonalization property of the noise covariance matrix, in the complex wavelet packet basis, has to be proven. The existence of a basis of continuous complex wavelets packets should also be investigated.

Structure of the tree

The structure of the wavelet packet decomposition tree should adapt to each problem, to improve the separation between the signal and the noise, while optimizing the speed. If the level of the noise is very high, the depth of the wavelet tree has to be increased, and the wavelet packet spaces can be split into different ways to enable the recovery of more details. For example, if needed, the lowest frequency subbands of the packet subspaces could be split instead of the highest frequency subbands. If the level of the noise is low,

the highest frequencies of the packet subbands become recoverable, therefore they can be filtered instead of being simply removed.

Finally, the tree should adapt to the statistics of the deconvolved noise, which are directly linked to the Fourier transform of the convolution kernel. An automatic method has to be found, which minimizes both the spatial localization of the wavelets and the correlation of the noise within the subbands. An hybrid tree [29] could be used, enabling decomposition of a subspace only along a single direction. This approach could better process the problems related to motion blur.

Subband modeling

The proposed modeling of the complex subbands could be improved and generalized to other types of images. For the homogeneous Generalized Gaussian model, a joint estimation of the exponent of the prior law and of the prior parameter should be studied.

An improvement of the adaptive Gaussian model could be found by choosing a more accurate law, to capture the heavy-tailed distribution of the wavelet packet coefficients. For example, adaptive Generalized Gaussian models can be investigated. However, the additional parameter (exponent of the law) has to be taken into account in the estimation step. Another possible modeling is to add a hidden variable, describing a hidden state related to each coefficient. It can be used to build a Gaussian mixture model, which marginal is heavy tailed, but which could make the estimation easier.

Interaction between scales

All of the previous types of modeling do not take into account the interaction between the different scales. The proposed technique filters each scale and each subband independently ; including the dependence between the different scales, by means of multiscale hidden Markov trees, could perhaps enable better separation of the small features from the deconvolved noise. These types of models certainly would better model the edges, which propagate through scales, and enable to reconstruct them more efficiently.

Interaction between coefficients

We have supposed that all the coefficients of a given subband are independent. It is not precisely the case for noise coefficients, because the diagonalization of the noise covariance matrix, provided by the packet transform, is not perfect. Therefore, it should be verified if a better denoising is provided by taking into account the dependence between noise variables.

We have supposed that the original signal coefficients are also independent. This is a very efficient approximation, since the computing time related to the thresholding step is very small, but it could be preferable to include the dependence between neigbour coefficients to correctly represent the subbands. Indeed, in an homogeneous area, the coefficients

are strongly correlated (they are constant). Along an edge, it is also true. Markov Random Fields (MRF) can efficiently capture such dependencies. Furthermore, by adding the inter-scale dependencies, it is possible to construct a hierarchical MRF [22]. The difficulty of such a model comes from the estimation, which could be almost intractable by classical methods, especially because of the adaptive nature of the model.

Choice of the best algorithm

We have seen, in the last chapter, that many methods can be used to threshold wavelet coefficients. Furthermore, there are many possible ways to combine wavelet thresholding with existing deconvolution techniques, such as nonlinear regularization. We have proposed two types of algorithms, one only based on thresholding, and the other one is hybrid since it combines two different approaches.

It is possible to derive new algorithms, by studying for example the possibility of injecting the result of DEPA [28] in the thresholding step (adaptive deconvolution in the spatial domain) instead of the result of RHEA [25], which is not adaptive. But DEPA uses an image obtained by wavelet thresholding to estimate the parameters of the model. Thus, we could construct an iterative method, alternating betweeen spatial and wavelet adaptive parameter estimation. The complexity of the study (convergence, robustness...) and the difficulty of implementation (choice of the elementary processing modules) suggest that the investigation should be done automatically. In such a case, a library of basic algorithms (thresholding, estimation...) is required, as well as a quality criterion, which should be able to decide which combination provides the best results.

Bibliography

[1] F. Abramovitch and T. Sapatinas. *Bayesian inference in wavelet based methods*, chapter Bayesian denoising of visual images in the wavelet domain. Springer, 1999.

[2] M. Antonini, M. Barlaud, P. Mathieu, and I. Daubechies. Image coding using wavelet transform. *IEEE Trans. on IP*, 1(2), Apr. 1992.

[3] G. Aubert and L. Vese. A variational method in image recovery. *SIAM Journal of Num. Anal.*, 34(5):1948–1979, Oct. 1997.

[4] M. Belge and E. Miller. Wavelet domain image restoration using edge preserving prior models. In *ICIP*, Chicago, USA, Oct. 1998.

[5] J. Bernardo and A. Smith. *Bayesian Theory*. John Wiley and Sons, Chichester, UK, 1994.

[6] A. Bijaoui, J-L. Starck, and F. Murtagh. Multiresolution support applied to image filtering and restoration. *Comp. Vision, Graphics and Image Proc. A.*, 1995.

[7] B. Chalmond. Image restoration using an estimated Markov model. *Signal Processing*, 15:115–129, 1988.

[8] S.G. Chang and M. Vetterli. Spatial adaptive wavelet thresholding for image denoising. *ICIP*, Oct. 1997.

[9] P. Charbonnier. *Reconstruction d'image : régularisation avec prise en compte de discontinuités*. PhD dissertation, Laboratoire I3S, UPRES-A 6070 du CNRS, University of Nice-Sophia Antipolis, France, Sept. 1994.

[10] P. Charbonnier, L. Blanc-Féraud, G. Aubert, and M. Barlaud. Deterministic Edge-Preserving Regularization in Computed Imaging. *IEEE Trans. on IP*, 6(2):298–311, Feb. 1997.

[11] A. Cohen, I. Daubechies, and J.-C. Feauveau. Biorthogonal bases of compactly supported wavelets. *Commun. on Pure and Appl. Math.*, 45:485–560, 1992.

[12] R. Coifman and D. Donoho. Translation-invariant de-noising. Technical Report 475, Stanford University, May 1995.

[13] R. Coifman, Y. Meyer, and M. Wickerhauser. *Wavelet analysis and signal processing*, pages 153–178. John and Barlett B. Ruskai et al. Editors, 1992.

[14] I. Daubechies. The wavelet transform, time-frequency localisation and signal analysis. *IEEE Trans on IT*, 36:961–1005, 1990.

[15] G. Demoment. Image reconstruction and restoration : overview of common estimation structures and problems. *IEEE Trans. on ASSP*, 37(12):2024–2036, Dec. 1989.

[16] D. Donoho. Denoising by soft thresholding. *IEEE Trans on IT*, 41:613–627, 1995.

[17] D. Donoho and I. Johnstone. Ideal spatial adaptation via wavelet shrinkage. *Biometrika*, 81:425–455, 1994.

[18] D. Donoho, I. Johnstone, G. Kerkyacharian, and D. Picard. Wavelet shrinkage: asymptopia. *J. Roy. Stat. Soc.*, B 57(2):301–369, 1995.

[19] M. Figueiredo and R. Nowak. Bayesian wavelet-based image estimation using noninformative priors. In *Part of the SPIE Conf. on Mathematical Modeling, Bayesian Estimation, and Inverse Problems*, Denver, USA, Jul. 1999.

[20] D. Geman and G. Reynolds. Constrained restoration and the recovery of discontinuities. *IEEE Trans. on PAMI*, 14(3):367–383, March 1992.

[21] S. Geman and D. Geman. Stochastic Relaxation, Gibbs Distributions and the Bayesian Restoration of Images. *IEEE Trans. on PAMI*, 6(6):721–741, Nov. 1984.

[22] C. Graffigne, F. Heitz, P. Perez, F. Prêteux, M. Sigelle, and J. Zerubia. Hierarchical Markov random field models applied to image analysis : a review. In *Proc. SPIE Conference on Neural Morphological and Stochastic Methods in Image and Signal Processing*, San Diego (USA), July 1995.

[23] C. W. Helstrom. Image restoration by the method of least squares. *J. Opt. Soc. Amer.*, 1967.

[24] A.K. Jain. A sinusoidal family of unitary transforms. *IEEE Trans. on PAMI*, 1(4):356–365, Oct. 1979.

[25] A. Jalobeanu, L. Blanc-Féraud, and J. Zerubia. Estimation d'hyperparamètres pour la restauration d'images satellitaires par une méthode MCMCML. *INRIA Research Report*, 3469, Aug. 1998.

[26] A. Jalobeanu, L. Blanc-Féraud, and J. Zerubia. Hyperparameter estimation for satellite image restoration by a MCMCML method. In Springer, editor, *LNCS - EMMCVPR*, York, July 1999.

[27] A. Jalobeanu, L. Blanc-Féraud, and J. Zerubia. Restauration automatique d'images satellitaires par une méthode MCMC. In *GRETSI*, Vannes, Sept. 1999.

[28] A. Jalobeanu, L. Blanc-Féraud, and J. Zerubia. Estimation of adaptive parameters for satellite image deconvolution. *INRIA Research Report*, June 2000.

[29] J. Kalifa. *Restauration minimax et déconvolution dans une base d'ondelettes miroirs*. PhD thesis, École Polytechnique, May. 1999.

[30] J. Kalifa and S. Mallat. Restauration d'images par paquets d'ondelettes. In *GRETSI*, Grenoble, France, 1997.

[31] J. Kalifa and S. Mallat. Wavelet packet deconvolutions. Technical report, CMAP, Ecole Polytechnique, Palaiseau, France, 1998.

[32] J. Kalifa and S. Mallat. *Bayesian inference in wavelet based methods*, chapter Minimax restoration and deconvolution. Springer, 1999.

[33] N. Kingsbury. The dual-tree complex wavelet transform: a new efficient tool for image restoration and enhancement. In *EUSIPCO*, pages 319–322, Rhodes, Greece, 1998.

[34] N. Kingsbury. The dual-tree complex wavelet transform: a new technique for shift invariance and directional filters. In *8th IEEE DSP Workshop*, Bryce Canyon UT, USA, 1998.

[35] N. Kingsbury. Image processing with complex wavelets. In *Phil. Trans. Roy. Soc. London A*, volume 357, pages 2543–2560, Special issue for the discussion meeting on "Wavelets: the key to intermittent information?" (held Feb. 24-25, 1999), Sept. 1999.

[36] M. Malfait and D. Roose. Wavelet-based image denoising using a markov random field a priori model. *IEEE Trans. on IP*, 6:549–565, 1997.

[37] S. Mallat. A theory for multiresolution signal decomposition: the wavelet representation. *IEEE Trans. on PAMI*, 11(7):674–693, Jul. 1989.

[38] S. Mallat. *A Wavelet tour of signal processing*. Academic Press, 1998.

[39] P. Moulin. A wavelet regularization method for diffuse radar-target imaging and speckle-noise reduction. *Journal of Mathematical Imaging and Vision*, 3(1):123–134, 1993.

[40] P. Moulin and J. Liu. Analysis of multiresolution image denoising schemes using generalized gaussian and complexity priors. *IEEE trans. on IT*, 45(3):909–919, Apr. 1999.

[41] C. Robert. *Méthodes de Monte Carlo par chaînes de Markov*. Statistique mathématique et probabilité. Economica, 1996.

[42] Bart M. Ter Haar Romeny, editor. *Geometry-driven diffusion in computer vision*. Kluwer Academic Publishers, 1994.

[43] E. Simoncelli. *Bayesian inference in wavelet based methods*, chapter Bayesian denoising of visual images in the wavelet domain. Springer, 1999.

[44] A.N. Tikhonov. Regularization of incorrectly posed problems. *Sov. Math. Dokl.*, 4:1624–1627, 1963.

[45] M. Vetterli. Filter banks allowing perfect reconstruction. *Signal Processing*, 10(3):219–244, 1986.

[46] M. Vetterli and J. Kovacevic. *Wavelets and subband coding*. Prentice Hall, 1995.

[47] M. Wickerhauser. *Adapted wavelet analysis from theory to software*. AK Peters, 1994.

Appendix A

Real wavelet transforms

A.1 Orthogonal wavelets

A.1.1 Unidimensional case

A wavelet ψ is a function of $\mathbf{L}^2(\mathbb{R})$ of 0 average, such as $||\psi||^2 = 1$, nearly centered on 0. This wavelet, scaled by 2^j and translated by $2^j n$, generates an orthonormal basis of $\mathbf{L}^2(\mathbb{R})$:

$$\left\{ \psi_{j,n}(t) = \frac{1}{\sqrt{2^j}} \psi \left(\frac{t - 2^j n}{2^j} \right) \right\}_{(j,n)\in\mathbb{Z}^2} \tag{A.1}$$

A multiresolution approximation [37] of a function is defined by a scaling function ϕ, which generates an orthonormal basis of V_j :

$$\left\{ \phi_{j,n}(t) = \frac{1}{\sqrt{2^j}} \phi \left(\frac{t - 2^j n}{2^j} \right) \right\}_{n\in\mathbb{Z}} \tag{A.2}$$

where V_j is a closed subspace of $\mathbf{L}^2(\mathbb{R})$ corresponding to the resolution 2^{-j}. ϕ has to satisfy particular conditions to ensure that each subspace V_j verifies all the properties of a multiresolution approximation [37].

A signal f can also be decomposed, for a given scale j, into an approximation part a and a detail part d :

$$a_n^j = \langle f(t), \phi_{j,n}(t) \rangle \qquad \text{and} \qquad d_n^j = \langle f(t), \psi_{j,n}(t) \rangle \tag{A.3}$$

There is a discrete filter h, such that the scaling functions ϕ are fully determined by h :

$$\frac{1}{\sqrt{2}} \phi(\frac{t}{2}) = \sum_{-\infty}^{+\infty} h_n \, \phi(t - n) \tag{A.4}$$

if the Fourier transform $\mathcal{F}[h]$ verifies the following conditions :

$$\forall \omega \in \mathbb{R}, \quad |\mathcal{F}[h](\omega)|^2 + |\mathcal{F}[h](\omega + \pi)|^2 = 2 \,, \quad |\omega| < \frac{\pi}{2} \Rightarrow |\tilde{h}(\omega)| > 0 \,, \quad \mathcal{F}[h](0) = \sqrt{2} \tag{A.5}$$

Then, h is said to be a conjugate mirror filter.

There is a discrete filter g, such that the wavelet function ψ is fully determined by g and the scaling function ϕ. Thus, the function ψ is determined by its Fourier transform :

$$\mathcal{F}[\psi](\omega) = \frac{1}{\sqrt{2}}\,\mathcal{F}[g](\frac{\omega}{2})\,\mathcal{F}[\phi](\frac{\omega}{2}) \tag{A.6}$$

which is equivalent to define ψ in the spatial domain by the following scaling relation :

$$\frac{1}{\sqrt{2}}\psi(\frac{t}{2}) = \sum_{-\infty}^{+\infty} g_n\,\phi(t-n) \tag{A.7}$$

The wavelet ψ generates an orthogonal basis of $\mathbf{L}^2(\mathbb{R})$ if the following sufficient condition between the discrete filters h and g is satisfied :

$$g_n = (-1)^{1-n} h_{1-n} \tag{A.8}$$

Discrete signals and discrete bases

As we deal with discrete signals, which are defined in the discrete space $\mathrm{l}^2(\mathbb{R})$, we use a filter bank to compute the wavelet transform [38, 46]. Decomposing a discrete signal consists of calculating scalar products with vectors of an orthonormal basis of $\mathrm{l}^2(\mathbb{R})$. The cascading of filters over J levels corresponds to the following basis :

$$\Big[\{\phi_{J,n}(k)\}_{n\in\mathbb{Z}}\,,\;\{\psi_{j,n}(k)\}_{1\le j\le J,\,n\in\mathbb{Z}}\Big]_{k\in\mathbb{Z}} \tag{A.9}$$

which is an orthonormal basis of $\mathrm{l}^2(\mathbb{R})$. It provides an orthogonal wavelet representation of $a^0 = f$, which consists of an approximation a^J at resolution 2^J and wavelet coefficients d^j, or details, at resolutions $1 \le 2^j \le 2^J$.

The discrete wavelet decomposition is performed by a fast filter bank technique :

$$a_n^{j+1} = (a^j \star \bar{h})_{2n} \qquad \text{and} \qquad d_n^{j+1} = (a^j \star \bar{g})_{2n} \tag{A.10}$$

where $(\,)_{2n}$ denotes a decimation (keeping only the coefficients having an odd index), and $\bar{h}_n = h_{-n}$. The approximation a and the detail d at scale $j+1$ are computed by convolving the approximation at scale j with $\bar{h}$ and $\bar{g}$ and decimating the result by a factor 2. We have $a^0 = f$.

Vetterli gives in [45, 46] sufficient conditions to ensure a perfect reconstruction when using such filter banks. These conditions are verified, as we have (A.8) and (A.5). Thus, the inverse transform is also achieved by a filter bank algorithm :

$$a_n^j = (\check{a}^{j+1} \star h)_n + (\check{d}^{j+1} \star g)_n \tag{A.11}$$

where $\check{a}_n$ is equal to a_p for $n = 2p$ and 0 otherwise. The reconstruction is an interpolation achieved by inserting zeroes followed by filtering.

A.1.2 Bidimensional case

A separable two-dimensional multiresolution is composed of the tensor product of two approximation spaces V_j and V_l :

$$V_{jl} = V_j \otimes V_l \tag{A.12}$$

and the basis generated by the separable scaling function $\phi(x)\phi(y)$:

$$\left\{ \phi_{j,l,n,m}(x,y) = \frac{1}{\sqrt{2^j 2^l}} \phi\left(\frac{x-2^j n}{2^j}\right) \phi\left(\frac{y-2^l m}{2^l}\right) \right\}_{(n,m)\in\mathbb{Z}^2} \tag{A.13}$$

is an orthonormal basis of V_{jl}.

Let ϕ be a scaling function and ψ the corresponding wavelet. Then the wavelet family :

$$\{\phi_{j,n}(x)\psi_{j,m}(y)\ ,\ \psi_{j,n}(x)\phi_{j,m}(y)\ ,\ \psi_{j,n}(x)\psi_{j,m}(y)\}_{(j,n,m)\in\mathbb{Z}^3} \tag{A.14}$$

is an orthonormal basis of $\mathbf{L}^2(\mathbb{R}^2)$.

We define the following separable wavelets :

$$\begin{cases} \psi^1_{j,n,m}(x,y) = \phi_{j,n}(x)\psi_{j,m}(y) \\ \psi^2_{j,n,m}(x,y) = \psi_{j,n}(x)\phi_{j,m}(y) \\ \psi^3_{j,n,m}(x,y) = \psi_{j,n}(x)\psi_{j,m}(y) \end{cases} \tag{A.15}$$

A function f of $\mathbf{L}^2(\mathbb{R}^2)$ is decomposed, for a given scale j, into an approximation part a and 3 detail parts d^1, d^2, d^3 :

$$a^j_{n,m} = \langle f(x,y), \phi_{j,n,m}(x,y)\rangle \qquad \text{and} \qquad d^{j,r}_{n,m} = \langle f(x,y), \psi^r_{j,n,m}(x,y)\rangle \tag{A.16}$$

Images and discrete bases

Decomposing a discrete image f of $\mathrm{l}^2(\mathbb{R}^2)$ consists of computing scalar products with vectors of an orthonormal basis of $\mathrm{l}^2(\mathbb{R}^2)$. The cascading of filters over J levels corresponds to the following basis :

$$\left[\{\phi_{J,n,m}(k,l)\}_{(n,m)\in\mathbb{Z}^2}\ ,\ \{\psi^r_{j,n,m}(k,l)\}_{1\le j\le J,\,(n,m)\in\mathbb{Z}^2,\,r\in\{1,2,3\}} \right]_{(k,l)\in\mathbb{Z}^2} \tag{A.17}$$

which is an orthonormal basis of $\mathrm{l}^2(\mathbb{R}^2)$. It provides an orthogonal wavelet representation of $a^0 = f$, which consists of an approximation a^J at resolution 2^J and wavelet coefficients $d^{j,1}, d^{j,2}, d^{j,3}$, or details, at resolutions $1 \le 2^j \le 2^J$.

Therefore, as in the unidimensional case, a fast algorithm can be used to achieve the discrete bidimensional wavelet transform :

$$\begin{aligned} a^{j+1}_{n,m} &= (a^j \star \bar{h}\bar{h})_{2n,2m} & d^{j+1,1}_{n,m} &= (a^j \star \bar{g}\bar{h})_{2n,2m} \\ d^{j+1,2}_{n,m} &= (a^j \star \bar{h}\bar{g})_{2n,2m} & d^{j+1,3}_{n,m} &= (a^j \star \bar{g}\bar{g})_{2n,2m} \end{aligned} \tag{A.18}$$

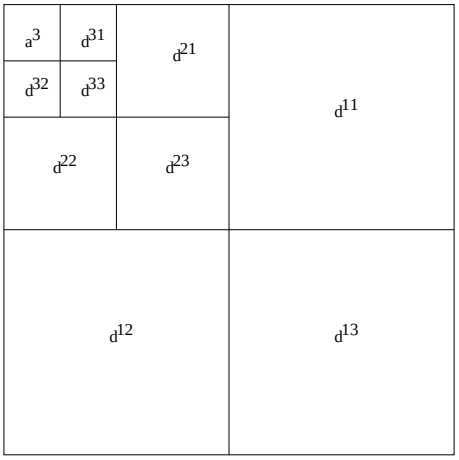
where $a \star uv$ denotes the convolution of a with the separable filter uv (convolution with u w.r.t. the columns and then with v w.r.t. the lines).

The inverse transform algorithm is given by :

$$a^j_{n,m} = (\check{a}^{j+1} \star hh)_{n,m} + (\check{d}^{j+1,1} \star gh)_{n,m} + (\check{d}^{j+1,2} \star hg)_{n,m} + (\check{d}^{j+1,3} \star gg)_{n,m} \quad \text{(A.19)}$$

The filters $xy_{m,n}$ are separable, i.e. $xy_{m,n} = x_m\, y_n$. The separable bidimensional convolutions can be factored into one-dimensional convolutions along rows and columns of the image. The results is subsampled by a factor 2, so the size of the resulting data is the same as the size of the initial approximation a^j. The corresponding approximate partition of the frequency plane and the wavelet transform are illustrated in Fig. A.1.

a)



b)

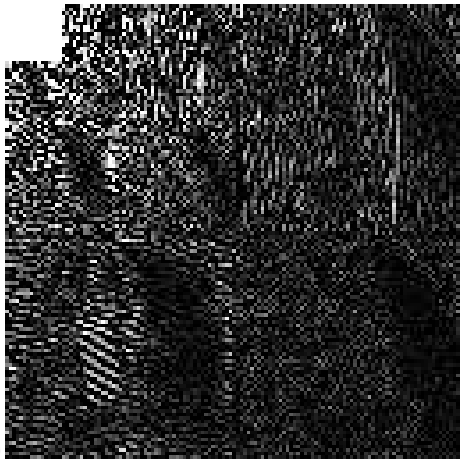


Figure A.1: *a) rough partition of the frequency space introduced by the dyadic wavelet transform, b) wavelet transform of the original image of Fig. 1.1*

A.2 Biorthogonal wavelets

A.2.1 Unidimensional case

If we require the wavelet ψ to be symmetric, or to have linear phase preservation which is very important in image processing, the wavelet basis $\{\psi_{j,m}(t)\}$ of equ. (A.1) cannot be orthogonal. It is possible to design linear phase wavelets with compact support and sufficient regularity by constructing biorthogonal wavelet bases of $\mathbf{L}^2(\mathbb{R})$ [11, 37].

It consists of constructing a new basis, dual of the first one. The first basis is used for the decomposition, and the second one for the reconstruction. These two bases are said to be biorthogonal [2].

We construct the basis $\{\tilde{\psi}_{j,m}(t)\}$, which is the dual basis of $\{\psi_{j,m}(t)\}$. The biorthogonality condition is expressed as :

$$\forall (j,m,j',m') \in \mathbb{Z}^4,\ \langle \tilde{\psi}_{j,m}(t)\,,\ \psi_{j',m'}(t) \rangle = \delta_{j,j'}\, \delta_{m,m'} \quad \text{(A.20)}$$

If the conditions of the Cohen-Daubechies-Feauveau theorem [11] are verified, then the families

$$\{\psi_{j,m}(t)\}_{(j,m)\in\mathbb{Z}^2} \quad \text{and} \quad \{\tilde{\psi}_{j,m}(t)\}_{(j,m)\in\mathbb{Z}^2} \tag{A.21}$$

are biorthogonal Riesz bases of $\mathbf{L}^2(\mathbb{R})$.

In the same manner, the families

$$\{\phi_{j,m}(t)\}_{m\in\mathbb{Z}} \quad \text{and} \quad \{\tilde{\phi}_{j,m}(t)\}_{m\in\mathbb{Z}} \tag{A.22}$$

are biorthogonal Riesz bases of V_j, the approximation space at scale 2^j.

Within this framework, different wavelets are used for analysis and synthesis, since there are two possible decompositions in these bases :

$$f(t) = \sum_{(j,m)\in\mathbb{Z}^2} \langle f(t)\,,\,\psi_{j,m}(t)\rangle \tilde{\psi}_{j,m}(t) = \sum_{(j,m)\in\mathbb{Z}^2} \langle f(t)\,,\,\tilde{\psi}_{j,m}(t)\rangle \psi_{j,m}(t) \tag{A.23}$$

A signal f can be decomposed, for a given scale j, into an approximation part a and a detail part d, as in the orthonormal case :

$$a_n^j = \langle f(t), \phi_{j,n}(t)\rangle \qquad \text{and} \qquad d_n^j = \langle f(t), \psi_{j,n}(t)\rangle \tag{A.24}$$

The functions ϕ and ψ are used for the analysis, and the functions $\tilde{\phi}$ and $\tilde{\psi}$ are used for the reconstruction.

The functions ψ, ϕ and the filters h, g satisfy to the equations (A.4), (A.6). On the other hand, the dual scaling functions $\tilde{\phi}$ are fully determined by a discrete filter $\tilde{h}$:

$$\frac{1}{\sqrt{2}}\tilde{\phi}(\frac{t}{2}) = \sum_{-\infty}^{+\infty} \tilde{h}_n\, \tilde{\phi}(t-n) \tag{A.25}$$

if the Fourier transform $\mathcal{F}[\tilde{h}]$ verifies the following conditions :

$$\begin{aligned} \forall\omega\in\mathbb{R}, \quad \mathcal{F}[h]^*(\omega)\mathcal{F}[\tilde{h}](\omega) + \mathcal{F}[h]^*(\omega+\pi)\mathcal{F}[\tilde{h}](\omega+\pi) = 2\,, \\ |\omega| < \frac{\pi}{2} \Rightarrow |\tilde{h}(\omega)| > 0\,, \quad \mathcal{F}[\tilde{h}](0) = \sqrt{2} \end{aligned} \tag{A.26}$$

The function $\tilde{\psi}$ is also determined by its Fourier transform :

$$\mathcal{F}[\tilde{\psi}](\omega) = \frac{1}{\sqrt{2}}\,\mathcal{F}[\tilde{g}](\frac{\omega}{2})\,\mathcal{F}[\tilde{\phi}](\frac{\omega}{2}) \tag{A.27}$$

The wavelets ψ and $\tilde{\psi}$ generate a biorthogonal basis of $\mathbf{L}^2(\mathbb{R})$ if the sufficient conditions between the discrete filters h, $\tilde{h}$, g and $\tilde{g}$ are verified :

$$g_n = (-1)^{1-n}\tilde{h}_{1-n} \quad \text{and} \quad \tilde{g}_n = (-1)^{1-n}h_{1-n} \tag{A.28}$$

Then, if we consider discrete signals, the following families of vectors generate a bi-orthogonal basis of $\mathrm{l}^2(\mathbb{R})$:

$$\left[\{\phi_{J,n}(k)\}_{n\in\mathbb{Z}}\,,\,\{\psi_{j,n}(k)\}_{1\leq j\leq J,\,n\in\mathbb{Z}}\right]_{k\in\mathbb{Z}} \tag{A.29}$$

$$\left[\{\tilde{\phi}_{J,n}(k)\}_{n\in\mathbb{Z}}\,,\,\{\tilde{\psi}_{j,n}(k)\}_{1\leq j\leq J,\,n\in\mathbb{Z}}\right]_{k\in\mathbb{Z}} \tag{A.30}$$

The first basis (A.29) provides a biorthogonal wavelet representation of $a^0 = f$, which consists of an approximation a^J at resolution 2^J and wavelet coefficients d^j, or details, at resolutions $1 \leq 2^j \leq 2^J$. The second basis (A.30) is needed for reconstruction.

The discrete wavelet decomposition is performed as in the orthogonal case by a fast filter bank technique :

$$a_n^{j+1} = (a^j \star h)_{2n} \qquad \text{and} \qquad d_n^{j+1} = (a^j \star g)_{2n} \tag{A.31}$$

Vetterli gives in [45, 46] sufficient conditions to ensure a perfect reconstruction when using such filter banks. These conditions are verified, as we have (A.28) and (A.26). Thus, the inverse transform is also achieved by a filter bank algorithm, but the dual filters $\tilde{h}$ and $\tilde{g}$ are needed :

$$a_n^j = (\check{a}^{j+1} \star \tilde{h})_n + (\check{d}^{j+1} \star \tilde{g})_n \tag{A.32}$$

Appendix B

Complex wavelet bases

B.1 Complex wavelets

Here, we consider the bidimensional complex wavelet transform defined by Algo. 3.1.3.

B.1.1 Generalization of biorthogonal bases

For the first tree A for example, in the unidimensional case, it is possible to generalize the discrete basis described by equations (A.29) and (A.30) by using, for each scale 2^j, wavelet functions generated by different mother wavelets ψ^o and ψ^e, corresponding to the odd and even filters h^o and h^e :

$$\left[\{\phi^e_{J,n}(k)\}_{n\in\mathbb{Z}}\,,\ \{\psi^o_{1,n}(k)\}_{n\in\mathbb{Z}}\,,\ \{\psi^e_{j,n}(k)\}_{2\le j\le J,\,n\in\mathbb{Z}}\right]_{k\in\mathbb{Z}} \tag{B.1}$$

$$\left[\{\tilde{\phi}^e_{J,n}(k)\}_{n\in\mathbb{Z}}\,,\ \{\tilde{\psi}^o_{1,n}(k)\}_{n\in\mathbb{Z}}\,,\ \{\tilde{\psi}^e_{j,n}(k)\}_{2\le j\le J,\,n\in\mathbb{Z}}\right]_{k\in\mathbb{Z}} \tag{B.2}$$

B.1.2 Separable biorthogonal bases

The following basis is a separable biorthogonal basis of $\mathbf{L}^2(\mathbb{R}^2)$, corresponding to the tree D of the complex wavelet transform :

$$\left\{\phi_{j,n}(x)\psi_{j,m}(y)\,,\ \psi_{j,n}(x)\phi_{j,m}(y)\,,\ \psi_{j,n}(x)\psi_{j,m}(y)\right\}_{(j,n,m)\in\mathbb{Z}^3} \tag{B.3}$$

$$\left\{\tilde{\phi}_{j,n}(x)\tilde{\psi}_{j,m}(y)\,,\ \tilde{\psi}_{j,n}(x)\tilde{\phi}_{j,m}(y)\,,\ \tilde{\psi}_{j,n}(x)\tilde{\psi}_{j,m}(y)\right\}_{(j,n,m)\in\mathbb{Z}^3} \tag{B.4}$$

if $\psi_{j,m}$ and $\tilde{\psi}_{j,m}$ generate a biothogonal basis of $\mathbf{L}^2(\mathbb{R})$.

If we consider a discrete image, it can be decomposed in the following separable discrete biorthogonal basis of $\mathrm{l}^2(\mathbb{R}^2)$:

$$\left[\begin{array}{l}\phi_{J,n}(k)\phi_{J,m}(l)\}_{(n,m)\in\mathbb{Z}^2}\,,\\ \{\phi_{j,n}(k)\psi_{j,m}(l)\,,\ \psi_{j,n}(k)\phi_{j,m}(l)\,,\ \psi_{j,n}(k)\psi_{j,m}(l)\}_{1\le j\le J,\,(n,m)\in\mathbb{Z}^2}\end{array}\right]_{(k,l)\in\mathbb{Z}^2} \tag{B.5}$$

$$\left[\begin{array}{l}\tilde{\phi}_{J,n}(k)\tilde{\phi}_{J,m}(l)\}_{(n,m)\in\mathbb{Z}^2}\,,\\ \{\tilde{\phi}_{j,n}(k)\tilde{\psi}_{j,m}(l)\,,\ \tilde{\psi}_{j,n}(k)\tilde{\phi}_{j,m}(l)\,,\ \tilde{\psi}_{j,n}(k)\tilde{\psi}_{j,m}(l)\}_{1\le j\le J,\,(n,m)\in\mathbb{Z}^2}\end{array}\right]_{(k,l)\in\mathbb{Z}^2} \tag{B.6}$$

B.2 Complex wavelet bases

For the trees A, B and C (see section 3.1.2) it is possible to generalize the discrete basis described by equations (B.5) and (B.6) by using, for each scale 2^j, separable wavelet functions generated by different mother wavelets ψ^o and ψ^e, corresponding to the odd and even filters h^o and h^g :

$$\left[\begin{array}{l} \{\phi^e_{J,n}(k)\phi^e_{J,m}(l)\}_{(n,m)\in\mathbb{Z}^2}\,, \\ \{\phi^o_{1,n}(k)\psi^o_{1,m}(l)\,,\ \psi^o_{1,n}(k)\phi^o_{1,m}(l)\,,\ \psi^o_{1,n}(k)\psi^o_{1,m}(l)\}_{m\in\mathbb{Z}}\,,\ \mathcal{W}_T \end{array} \right]_{(k,l)\in\mathbb{Z}^2} \tag{B.7}$$

$$\left[\begin{array}{l} \{\tilde{\phi}^e_{J,n}(k)\tilde{\phi}^e_{J,m}(l)\}_{(n,m)\in\mathbb{Z}^2}\,, \\ \{\tilde{\phi}^o_{1,n}(k)\tilde{\psi}^o_{1,m}(l)\,,\ \tilde{\psi}^o_{1,n}(k)\tilde{\phi}^o_{1,m}(l)\,,\ \tilde{\psi}^o_{1,n}(k)\tilde{\psi}^o_{1,m}(l)\}_{m\in\mathbb{Z}}\,,\ \tilde{\mathcal{W}}_T \end{array} \right]_{(k,l)\in\mathbb{Z}^2} \tag{B.8}$$

The family $\mathcal{W}_T$ is constructed by products of odd and even functions, corresponding to the 4 different trees. $\tilde{\mathcal{W}}_T$ is obtained in the same manner, by taking the dual functions.

$$\mathcal{W}_A = \{\phi^e_{j,n}(k)\psi^e_{j,m}(l)\,,\ \psi^e_{j,n}(k)\phi^e_{j,m}(l)\,,\ \psi^e_{j,n}(k)\psi^e_{j,m}(l)\}_{2\le j\le J,\,(n,m)\in\mathbb{Z}^2} \tag{B.9}$$

$$\mathcal{W}_B = \{\phi^e_{j,n}(k)\psi^o_{j,m}(l)\,,\ \psi^e_{j,n}(k)\phi^o_{j,m}(l)\,,\ \psi^e_{j,n}(k)\psi^o_{j,m}(l)\}_{2\le j\le J,\,(n,m)\in\mathbb{Z}^2} \tag{B.10}$$

$$\mathcal{W}_C = \{\phi^o_{j,n}(k)\psi^e_{j,m}(l)\,,\ \psi^o_{j,n}(k)\phi^e_{j,m}(l)\,,\ \psi^o_{j,n}(k)\psi^e_{j,m}(l)\}_{2\le j\le J,\,(n,m)\in\mathbb{Z}^2} \tag{B.11}$$

$$\mathcal{W}_D = \{\phi^o_{j,n}(k)\psi^o_{j,m}(l)\,,\ \psi^o_{j,n}(k)\phi^o_{j,m}(l)\,,\ \psi^o_{j,n}(k)\psi^o_{j,m}(l)\}_{2\le j\le J,\,(n,m)\in\mathbb{Z}^2} \tag{B.12}$$

B.3 Complex wavelet packets

Here, we consider the bidimensional Complex Wavelet Packet Transform (CWPT) defined by Algo. 3.2.1.

B.3.1 Complex wavelet packet bases

To construct the bases associated to the wavelet packets, the scaling relations (2.10) and (2.11) are used, for each type of wavelet ψ^o or ψ^e, as in the real case. We then obtain new functions $\psi^{o,p}$ and $\psi^{e,p}$ corresponding to the wavelet packets. The dual functions are also needed for reconstruction ; they are defined in the same manner.

Since it is difficult to define a basis of continuous wavelets which corresponds to the transform defined by algorithm 3.2.1, we will only define the discrete basis associated to each tree. An hybrid biorthogonal basis is defined as for the complex wavelet transform.

Let $\mathbf{Q}$ be an admissible quad-tree in the sense of equ. (2.8). $\mathbf{Q}$ is used to construct the 4 trees A, B, C, D. Finally, a discrete image is decomposed in the orthonormal basis of $\mathrm{l}^2(\mathbb{R}^2)$ defined by equations (B.7) and (B.8), but the families $\mathcal{W}_T$ have to be replaced by the packet families $\mathcal{W}^P_T$ defined in the following equations, where $\mathbf{Q}_{2\le j\le J}$ is an admissible quad-tree

of maximum depth J, with $j \geq 2$:

$$\mathcal{W}_A^P = \{\phi_{j,n}^{e,p}(k)\psi_{j,m}^{e,q}(l)\}_{(n,m)\in\mathbb{Z}^2,(j,p,q)\in\mathbf{Q}_{2\leq j\leq J}} \quad \text{(B.13)}$$

$$\mathcal{W}_B^P = \{\phi_{j,n}^{e,p}(k)\psi_{j,m}^{o,q}(l)\}_{(n,m)\in\mathbb{Z}^2,(j,p,q)\in\mathbf{Q}_{2\leq j\leq J}} \quad \text{(B.14)}$$

$$\mathcal{W}_C^P = \{\phi_{j,n}^{o,p}(k)\psi_{j,m}^{e,q}(l)\}_{(n,m)\in\mathbb{Z}^2,(j,p,q)\in\mathbf{Q}_{2\leq j\leq J}} \quad \text{(B.15)}$$

$$\mathcal{W}_D^P = \{\phi_{j,n}^{o,p}(k)\psi_{j,m}^{o,q}(l)\}_{(n,m)\in\mathbb{Z}^2,(j,p,q)\in\mathbf{Q}_{2\leq j\leq J}} \quad \text{(B.16)}$$

Appendix C

Covariance of the deconvolved noise

The noise N is supposed to be white, Gaussian and stationary, of variance σ^2. Then we have :

$$E\left[N\,N^t\right] = \sigma^2 I$$

After deconvolution by the operator H, which is block circulant, and supposed to be invertible, we have $N_d = H^{-1}N$:

$$E\left[H^{-1}N\,N^t H^{-1\,t}\right] = \sigma^2 H^{-1}H^{-1\,t}$$

The noise remains stationary, since it is a stationary noise convolved by a shift invariant inverse filter. Indeed, $H^{-1}H^{-1\,t}$ is block-circulant, and the diagonal elements are equal.

After deconvolution, the noise is *colored* (the covariance matrix is not diagonal anymore).

After projection in the subband k of a real or complex wavelet transform, without decimation, the noise is stationary and colored. We call σ_k^2 the variance of the real or imaginary part of the noise coefficients in the subband k, i.e. the diagonal element of the corresponding covariance matrix :

$$\sigma^2 W^k H^{-1} H^{-1\,t} W^{k\,t}$$

where W^k is the block circulant operator corresponding to the projection in the subband k.

To compute σ^k, we compute the covariance in the Fourier domain, since the Fourier basis diagonalizes the block-circulant operators. Let F be the unitary matrix of the discrete Fourier transform. The covariance of the noise N after deconvolution by H^{-1} in the Fourier domain is :

$$E\left[FH^{-1}N\,N^t H^{-1\,t}F^t\right] = \sigma^2 FH^{-1}H^{-1\,t}F^t = \sigma^2 \mathrm{diag}\left\{\frac{1}{|\mathcal{F}[h]_{ij}|^2}\right\}$$

Indeed, H and H^{-1} are diagonalized by the Fourier transform and

$$FHF^t = \mathrm{diag}\left\{\mathcal{F}[h]_{ij}\right\}$$

$$FH^{-1}F^t = \mathrm{diag}\left\{\frac{1}{\mathcal{F}[h]_{ij}}\right\}$$

$$\text{and} \quad FH^{-1}H^{-1\,t}F^t = \text{diag}\left\{\frac{1}{|\mathcal{F}[h]_{ij}|^2}\right\}$$

The same properties remain valid for the decomposition using W^k, then we finally obtain :

$$E\left[FW^kH^{-1}N\,N^tH^{-1\,t}W^{k\,t}F^t\right] = \sigma^2\text{diag}\left\{\frac{|\mathcal{F}[W^k]_{ij}|^2}{|\mathcal{F}[h]_{ij}|^2}\right\}$$

By using Parseval's theorem, we finally obtain :

$$\sum_{i,j}\sigma_k^2 = \sum_{ij}\sigma^2\frac{|\mathcal{F}[W^k]_{ij}|^2}{|\mathcal{F}[h]_{ij}|^2}$$

which finally gives :

$$\sigma_k^2 = \frac{1}{N_xN_y}\sum_{ij}\sigma^2\frac{|\mathcal{F}[W^k]_{ij}|^2}{|\mathcal{F}[h]_{ij}|^2}$$

Appendix D

Computation of the residual noise

D.1 Energy of the residual noise

We have to compute the variance of the residual noise after thresholding the CWPT coefficients, but we also must avoid slowing down the deconvolution algorithm. Therefore, we cannot simply generate one or more noise images of variance σ^2, and then apply the COWPATH algorithm to it and compute the variance of the resulting images.

For the homogeneous thresholding methods of Section 3.3.3 and for the method described in Section 3.3.4, we will show a general computation scheme able to evaluate the residual variance $(\sigma^r)^2$, by evaluating simple expressions, only taking into account σ_k and α_k. However, the most efficient method, described in Section 3.3.5, requires a more complicated computation, but it can also be achieved without slowing down the deconvolution process.

D.1.1 Homogeneous thresholding

So, a simple way to compute the variance is to compute the energy of the resulting image, since we assume the noise is zero-mean. This energy is conserved by the inverse CWPT. Therefore it can be simply obtained by summing the squared magnitudes of the thresholded coefficients, in each subband separately. We then consider the calculus of this energy for a given subband k. To obtain an estimate of the global variance, we have to divide the energy by the number of pixels of the image, $N_x N_y$ (we assume $N_x N_y \gg 1$). Let us denote n_{ij}^k the coefficient (i, j), subband k, of the CWPT of the deconvolved noise. Then we have :

$$(\sigma^r)^2 \simeq \frac{1}{N_x N_y} \sum_k \sum_{ij} |\theta(n_{ij}^k)|^2 \tag{D.1}$$

Instead of computing it from simulated and deconvolved noise, we can take the exact expression of the variance, which is simply given by taking the expectation of this expression, w.r.t. the law of the noise n_{ij}^k, which is normally distributed with zero mean and variance σ_k^2. We suppose that the noise coefficients are decorrelated in the wavelet packet basis, since this basis nearly diagonalizes the noise covariance matrix.

Then, the sum over the coefficients reduces to multiply the variance of the thresholded noise by N_k, the number of coefficients of the subband k :

$$(\sigma^r)^2 = \frac{1}{N_x N_y} \sum_k N_k \, E\left[|\theta(N(0,\sigma_k^2))|^2\right] \tag{D.2}$$

Thus, we have to compute $(\sigma_k^r)^2$, the residual variance in the subband k :

$$(\sigma_k^r)^2 = E\left[|\theta(N(0,\sigma_k^2))|^2\right] \tag{D.3}$$

which can be evaluated by calculating the following integral :

$$(\sigma_k^r)^2 = \frac{1}{2\pi\sigma_k^2} \int_{x\in\mathbb{C}} e^{-|x|^2/2\sigma_k^2} |\theta(x)|^2 \, dx \tag{D.4}$$

Since θ can be written as an attenuation function, only taking into account the magnitude of the coefficient to preserve shift invariance, i.e. $\theta(x) = a(|x|)x$, this integral can be computed with real numbers :

$$(\sigma_k^r)^2 = \frac{1}{\sigma_k^2} \int_0^\infty e^{-r^2/2\sigma_k^2} (a(r))^2 \, r^3 \, dr \tag{D.5}$$

which can be computed numerically.

The function θ depends on the thresholding method. If the threshold T only depends on σ_k, and is proportional to σ_k, equ. (D.5) simplifies and gives $(\sigma_k^r)^2 = K\,\sigma_k^2$, where K is a constant which only depends on the thresholding method. This is the case for hard and soft thresholding with $T \propto \sigma_k$, but also for the noninformative prior method of Section 3.3.4.

If the T depends also on a prior parameter α_k, the integral (D.5) must be computed for each subband, since it also depends on α_k.

D.1.2 Adaptive thresholding

The residual noise related to the adaptive method of Section 3.3.5 is more difficult to compute, since in the constant areas it is linked to the approximate original image used for the oracle thresholding. The approximate original image exhibits some noise in these areas, with real or imaginary parts of variance $\tilde{\sigma}_k^2$, given by equ. (3.48), which could induce insufficient noise filtering of the CWPT coefficients, if used as is. That is why the CWPT of this approximate image is also filtered, before being used in the attenuation factor, in order to reduce as much as possible the residual noise. Finally, the residual noise variance is a function of σ_k and $\tilde{\sigma}_k$, but also of the correlation c_k between the noise of the two transforms.

Covariance of both deconvolved noises

To compute the covariance c_k of the two types of deconvolved noise N_b and N_d (see section 3.3.7), we use the same approach as for computing the variance. We compute the mean of

the product $N_d N_b$ of the two noises, which can be estimated by summing these products over the whole image, and then we divide it by the number of coefficients. This sum is equal to the $(0,0)$ coefficient of the Fourier transform of the product :

$$c_k \simeq \frac{1}{N_x N_y} \sum_{ij} (N_d N_b)_{ij} = \frac{1}{N_x N_y} \mathcal{F}[N_d N_b]_{00} = \frac{1}{N_x N_y} \left(\mathcal{F}[N_d] \star \mathcal{F}[N_b]^*\right)_{00} \quad \text{(D.6)}$$

We only need to evaluate the $(0,0)$ coefficient of the convolution term $\mathcal{F}[N_d] \star \mathcal{F}[N_b]^*$. By taking into account the expressions (3.45) and (3.47), we obtain :

$$c_k \simeq \sigma^2 \frac{1}{N_x N_y} \sum_{ij} \frac{|\mathcal{F}[R^k]_{ij}|^2}{|\mathcal{F}[h]|^2_{ij} + 2\sigma^2 b(|\mathcal{F}[d_x]|^2_{ij} + |\mathcal{F}[d_y]|^2_{ij})} \quad \text{(D.7)}$$

Residual noise

We denote a_o the oracle attenuation function, related to the oracle thresholding function (3.44), such as $\theta_o(x) = a_o(x)\, x$, and θ_J the noninformative thresholding function (3.36). a_o multiplies the deconvolved noise coefficients n^k_{ij}, the ground truth is contaminated by a Gaussian noise $\tilde{n}^k_{ij}$, also normally distributed, whose real and imaginary parts variance is $\tilde{\sigma}^2_k$. Then, equ. (D.1) must be replaced by :

$$(\sigma^r)^2 \simeq \frac{1}{N_x N_y} \sum_k \sum_{ij} \left| n^k_{ij} a_o \left(\theta_J(\tilde{n}^k_{ij})\right)\right|^2 \quad \text{(D.8)}$$

which gives the residual variance in each subband :

$$(\sigma^r_k)^2 = E\left[\left| n^k_{ij} a_o \left(\theta_J(\tilde{n}^k_{ij})\right)\right|^2\right] \quad \text{(D.9)}$$

which can be evaluated by calculating an integral, as seen previously. The complex variables x and y respectively denote CWPT coefficients of the deconvolved noise and of the noise in the ground truth. The joint density of x and y is written as :

$$P(x,y) = \frac{1}{Z} e^{-A|x|^2 - B|y|^2 - C(x_r y_r + x_i y_i)} \quad \text{(D.10)}$$

where A, B and C are linked to the variances and covariance :

$$A = \frac{\sigma^2_k}{2\Delta} \quad B = \frac{\tilde{\sigma}^2_k}{2\Delta} \quad C = -\frac{c_k}{\Delta} \qquad \text{with } \Delta = \sigma^2_k \tilde{\sigma}^2_k - c^2_k \quad \text{(D.11)}$$

Z is the normalizing constant of the joint probability of x and y ; we have $Z = 4\pi^2 \Delta$.

Then, we can rewrite equ. (D.9) as follows :

$$(\sigma^r_k)^2 = \frac{1}{Z} \int_{(x,y)\in\mathbb{C}^2} e^{-A|x|^2 - B|y|^2 - C(x_r y_r + x_i y_i)} |y|^2 \, a_o(\theta_J(x))^2 \, dx\, dy \quad \text{(D.12)}$$

Let us first integrate w.r.t. y. We have :

$$\int_{y\in\mathbb{C}} e^{-B|y|^2 - C(x_r y_r + x_i y_i)} |y|^2 \, dy = \frac{\pi}{B} e^{|x|^2 C^2/4B} (1 + |x|^2 C^2/4B) \tag{D.13}$$

Then we can integrate w.r.t. x :

$$(\sigma_k^r)^2 = \frac{1}{4\pi\Delta B} \int_{x\in\mathbb{C}} e^{-|x|^2/2\tilde{\sigma}_k^2} (1 + \beta|x|^2) \, a_o(\theta_J(x))^2 \, dx \tag{D.14}$$

$$\text{where} \quad \beta = \frac{C^2}{4B} = \frac{1}{2\tilde{\sigma}_k^2(\sigma_k^2\tilde{\sigma}_k^2/c_k^2 - 1)} \tag{D.15}$$

To simplify, we write $\theta_J(x) = a_J(|x|)x$, then we have

$$a_o(\theta_J(x))^2 = \left(1 + 2\frac{\sigma_k^2}{|x|^2 a_J(|x|)^2}\right)^{-2}$$

This enables us to calculate the integral only using real numbers, because the above expressions only depend on the magnitude $|x| = r$:

$$(\sigma_k^r)^2 = \frac{1}{\tilde{\sigma}_k^2} \int_0^\infty e^{-r^2/2\tilde{\sigma}_k^2} (r + \beta r^3) \, \left(1 + 2\frac{\sigma_k^2}{r^2 a_J(r)^2}\right)^{-2} \, dr \tag{D.16}$$

By taking the expression of $a_J(r)$, we obtain :

$$(\sigma_k^r)^2 = \frac{1}{\tilde{\sigma}_k^2} \int_0^\infty e^{-r^2/2\tilde{\sigma}_k^2} (r + \beta r^3) \, \left(1 + 2\frac{\sigma_k^2 r^2}{(r^2 - 4\tilde{\sigma}_k^2)_+^2}\right)^{-2} \, dr \tag{D.17}$$

So it is sufficient to know σ_k, $\tilde{\sigma}_k$ and c_k for each subband, to compute the integral (D.17) numerically, using equ. (D.15) for β.

Finally, the global residual variance, for all thresholding methods, is obtained after computing the variance in each subband (cf. equ. (D.2) and (D.3)) :

$$(\sigma^r)^2 = \frac{1}{N_x N_y} \sum_k N_k \, (\sigma_k^r)^2 \tag{D.18}$$

D.2 Gradients of the residual noise

If we want the results of the COWPATH algorithms to be used as a starting point of an adaptive parameter estimation method, such as the one described in [28], we also have to evaluate the variance of the gradients (i.e. adjacent pixel differences) of the residual noise. We no longer can use the previous approach, since the knowledge of the energy of this

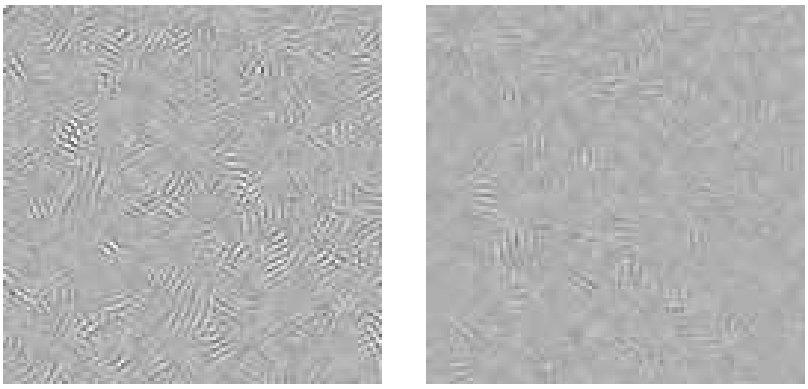


Figure D.1: *Examples of residual noise in homogeneous areas, amplified by a factor 8, related to the image of Fig. (4.1) : a) COWPATH 1, method 2 (noninformative prior) - $\sigma^r \simeq 2.7$, b) accelerated COWPATH 2 - $\sigma^r \simeq 0.8$*

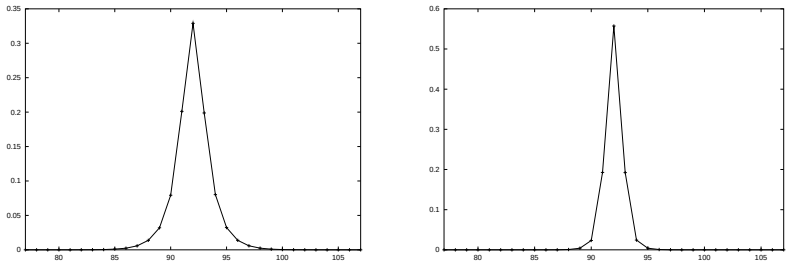


Figure D.2: *Examples of residual noise histograms in homogeneous areas, related to the image of Fig. (4.1) : a) COWPATH 1, method 2 (noninformative prior), b) accelerated COWPATH 2*

noise is not sufficient to determine also the energy of the noise gradients. Indeed, we need to know the covariance between adjacent pixels of the residual. There is no way to link the circular convolution operators related to the derivatives, with the CWPT operators (as in Section 3.3.7), because there is a nonlinear thresholding between them.

So we consider images containing only white noise of variance σ^2. We know the deconvolved noise variances in the different subbands given by Equ. (3.46) and (3.48), and the covariance given by Equ. (D.7). So we can simulate the noise by Gaussian variables. We simulate the coefficients independently, assuming that the covariance matrix is nearly diagonal in the chosen basis. We then threshold the simulated variables, and then study the distribution of the inverse CWPT of the result.

The resulting thresholded noise is no more Gaussian (even if its distribution has a Gaussian tail), and the filtered subbands are finally combined through the inverse CWPT to form the reconstructed residual noise, so there is no simple method to calculate this noise. Therefore, the computation is made by a Monte Carlo method [41]:

- Simulate the deconvolved noise in the subbands, taking into account the variances σ_k^2 and $\tilde{\sigma}_k^2$, and the covariance c_k ;
- Apply Algo. 3.4.2 to deconvolve this image. In the RHEA [25] algorithm, use the same parameters as for the data image (do not estimate them from the noise image). The result corresponds to the residual noise.
- Compute the gradients w.r.t. rows and columns, and estimate their variance by a sum of squares. Compute also the histogram to study the behaviour of the distribution function of the noise.

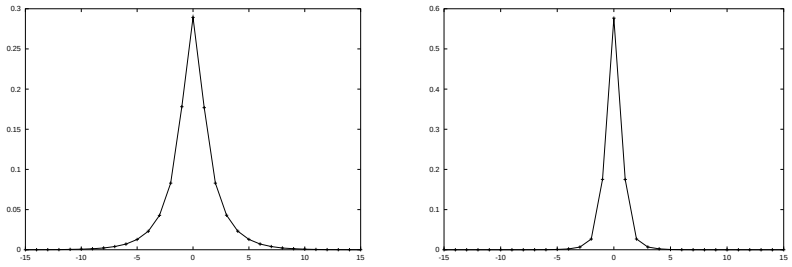


Figure D.3: *Examples of residual noise gradient histograms in homogeneous areas, related to the image of Fig. (4.1) : a) COWPATH 1, method 2 (noninformative prior), b) accelerated COWPATH 2*

Unité de recherche INRIA Sophia Antipolis
2004, route des Lucioles - B.P. 93 - 06902 Sophia Antipolis Cedex (France)

Unité de recherche INRIA Lorraine : Technopôle de Nancy-Brabois - Campus scientifi que
615, rue du Jardin Botanique - B.P. 101 - 54602 Villers lès Nancy Cedex (France)
Unité de recherche INRIA Rennes : IRISA, Campus universitaire de Beaulieu - 35042 Rennes Cedex (France)
Unité de recherche INRIA Rhône-Alpes : 655, avenue de l'Europe - 38330 Montbonnot St Martin (France)
Unité de recherche INRIA Rocquencourt : Domaine de Voluceau - Rocquencourt - B.P. 105 - 78153 Le Chesnay Cedex (France)

Éditeur
INRIA - Domaine de Voluceau - Rocquencourt, B.P. 105 - 78153 Le Chesnay Cedex (France)
http://www.inria.fr
ISSN 0249-6399